



Estimation circulaire multi-modèles appliquée au Map matching en environnement contraint

Karim El Mokhtari

► To cite this version:

Karim El Mokhtari. Estimation circulaire multi-modèles appliquée au Map matching en environnement contraint. Vision par ordinateur et reconnaissance de formes [cs.CV]. Université du Littoral Côte d'Opale; Université Abdelmalek Essaadi (Tanger, Maroc). Faculté des Sciences et Techniques, 2015. Français. NNT : 2015DUNK0367 . tel-01260019

HAL Id: tel-01260019

<https://theses.hal.science/tel-01260019>

Submitted on 21 Jan 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

CGR de Calais
École Doctorale Sciences pour l'Ingénieur
Numéro d'ordre :

Université Abdelmalek Essaâdi
Faculté des Sciences et Techniques
Tanger - Maroc

THÈSE EN COTUTELLE

présentée et soutenue publiquement le 08 janvier 2015

pour l'obtention du grade de

Docteur de l'Université du Littoral Côte d'Opale
et Docteur de l'Université Abdelmalek Essaâdi

(Discipline : Génie Informatique, Automatique et Traitement du Signal)

par

Karim EL MOKHTARI

Titre :

Estimation circulaire multi-modèles appliquée au Map matching en environnement contraint

Composition du jury

- Président :* **Mohammed BENJELLOUN**
Professeur à l'Université du Littoral Côte d'Opale, Calais, France
- Rapporteurs :* **Mohamed ESSAAIDI**
Professeur à l'ENSIAS, Université Mohammed V, Rabat, Maroc
Roland CHAPUIS
Professeur à l'Université Blaise-Pascal de Clermont-Ferrand, France
Salaheddine HAMDOUNE
Professeur habilité à la Faculté des Sciences et Techniques, Tanger, Maroc
- Directeurs de Thèse :* **Serge REBOUL**
Maître de Conférences HDR à l'Université du Littoral Côte d'Opale, Calais, France
Benaissa AMAMI
Professeur à la Faculté des Sciences et Techniques, Tanger, Maroc

*À la mémoire de mon père qui a fait de moi l'homme que je suis,
À la personne qui m'a donné la vie et l'amour inconditionnel, ma mère,
À la femme qui règne sur mon cœur, mon épouse,
Aux trois petites étoiles qui illuminent mon ciel, mes enfants.*

Remerciements

Je voudrais adresser mes premiers remerciements à mon encadrant Monsieur Serge Reboul. Je ne saurais trouver les mots pour vous remercier de votre dévouement, votre disponibilité et votre support sans limite. Vous avez su me redonner "l'envie d'avoir envie", et rendre ce travail du vrai plaisir au lieu d'un lourd devoir. Vous avez ravivé en moi l'esprit de défi, de courage et de curiosité malgré la longue distance qui nous sépare au quotidien.

En ce même instant, je me rappelle du mois de juin 2006 et d'un grand nom dans l'histoire de cette thèse, Monsieur Mohammed Benjelloun qui m'avait invité à l'ULCO et était à l'origine de mon premier contact avec le monde de la recherche. Vous êtes aussi le Directeur de mes travaux en France pendant les premières années de cette thèse. Nul mot ne pourrait exprimer ma gratitude pour tous vos efforts, votre gentillesse inégalée et pour le grand honneur que vous me faites en présidant mon jury.

J'adresse mes sincères remerciements à Monsieur Benaissa Amami, mon Directeur de thèse au Maroc. Vous n'avez épargné aucun effort bien avant de commencer cette thèse pour m'encourager à me lancer dans ce domaine. Merci pour vos suggestions pertinentes, votre suivi et votre soutien que j'ai beaucoup apprécié le long de ces années.

Je tiens à remercier l'ensemble des membres de mon jury de thèse, je pense à Messieurs Mohamed Essaaidi, Roland Chapuis et Salaheddine Hamdoune par leur travail précis et constructif de rapporteurs. Je vous remercie pour avoir utilisé vos connaissances dans la lecture fine de mon manuscrit.

Je ne pourrais terminer cette page de remerciements sans citer plusieurs noms, je pense notamment à Jean-Bernard Choquel pour sa collaboration enrichissante et ses retouches minutieuses, à Florian Beaubois pour son aide dans les expérimentations. Un remerciement fraternel à des noms qui me sont chers Monir Azmani, Georges Stienne et Abderrahim Bourkane. Merci pour les excellents moments partagés à Calais et au Maroc. J'adresse aussi ma reconnaissance à l'ensemble des personnes qui ont contribué de loin ou de près à la réussite de ce travail.

Je n'oublie pas dans les dernières lignes de cette page d'adresser mon remerciement et mon amour à mon épouse pour son affection, son soutien inconditionnel et surtout sa patience pendant mes longues heures de travail. Merci à mes chers parents pour l'éducation qu'ils m'ont apportée et l'amour et l'affection qu'ils m'ont offert.

Karim El mokhtari

Table des matières

Remerciements	iii
Glossaire	ix
Abréviations	xiii
Introduction générale	1
Chapitre 1 - Map matching	5
1.1 Introduction	5
1.2 Position du problème	6
1.3 Analyse géométrique	7
1.3.1 Map matching « point-to-point »	7
1.3.2 Map matching « point-to-curve »	7
1.3.3 Map matching « curve-to-curve »	10
1.3.4 Matching de surface (« <i>Area matching</i> »)	13
1.4 Analyse topologique	14
1.4.1 Algorithme topologique	15
1.4.2 Algorithme topologique amélioré	17
1.5 Analyse probabiliste	24
1.5.1 Détermination de la zone de confiance autour d'un point GPS	26
1.5.2 Déroulement de l'algorithme de Map matching probabiliste	28
1.5.3 Détermination de la position du véhicule sur le lien	29
1.5.4 Calcul de la variance des erreurs	32
1.5.5 Algorithme détaillé de Map matching reposant sur l'approche probabiliste	34
1.6 Autres approches d'analyse	36
1.6.1 Analyse par logique floue	36
1.6.2 Analyse par la théorie des croyances	38

1.6.3	Analyse par l'utilisation d'une carte précise	39
1.6.4	Analyse locale et globale	40
1.7	Conclusion	41
Chapitre 2	Filtrage multi-modèles	43
2.1	Introduction	43
2.2	Estimation et filtrage d'état	45
2.2.1	Estimateurs du maximum de vraisemblance et du maximum <i>a posteriori</i>	46
2.2.2	Estimateurs des moindres carrés et au sens des moindres carrés	48
2.2.3	Le filtre de Kalman	49
2.2.4	Méthodes de Monte Carlo : le filtre particulaire	54
2.3	Principe de l'approche multi-modèles	58
2.4	L'estimateur multi-modèles statique ou autonome	61
2.5	Les estimateurs multi-modèles dynamiques	65
2.5.1	Estimateur Pseudo-bayésien généralisé du premier ordre (GPB1)	66
2.5.2	Filtre Pseudo-bayésien généralisé du second ordre (GPB2)	67
2.5.3	Filtre multi-modèles interactif (IMM)	69
2.6	Estimateur multi-modèles à structure variable (VSMM)	75
2.6.1	L'approche d'adaptation récursive de l'ensemble des modèles (RAMS)	76
2.6.2	Décision sur un ensemble de modèles candidats	77
2.7	Conclusion	79
Chapitre 3	Map matching particulière	81
3.1	Introduction	81
3.2	Position du problème	82
3.2.1	Les limites des approches classiques	82
3.2.2	Localisation par l'estime ou <i>Dead Reckoning</i>	85
3.2.3	L'approche proposée	86
3.3	Map matching circulaire par filtrage particulaire SIR	88
3.3.1	Modélisation d'état	88
3.3.2	Modélisation circulaire	90
3.3.3	Implémentation particulière	92
3.3.4	Fonctionnement du filtre	95
3.4	Pré-filtrage circulaire	96
3.4.1	Architecture du système de préfiltrage	96

3.4.2	Pré-filtrage d'état circulaire	97
3.4.3	Pré-filtrage particulaire circulaire	100
3.5	Évaluation des méthodes	101
3.5.1	Introduction	101
3.5.2	Estimation de la direction	102
3.5.3	Estimation de la position	103
3.5.4	Identification de la route	106
3.6	Conclusion	110
Chapitre 4 - L'IMM circulaire		113
4.1	Introduction	113
4.2	Position du problème	115
4.2.1	Limites de l'approche particulaire circulaire	115
4.2.2	L'approche proposée	117
4.2.3	Les modèles circulaires	118
4.3	L'IMM circulaire	121
4.3.1	L'estimation circulaire	121
4.3.2	Le filtre IMM circulaire	124
4.4	Map matching circulaire particulaire avec pré-filtrage IMM	129
4.4.1	Architecture du système avec pré-filtrage IMM	129
4.4.2	Placement des particules par les probabilités de modèle	131
4.5	Évaluation de la méthode	132
4.5.1	Introduction	132
4.5.2	Estimation de la direction	133
4.5.3	Estimation de la position	134
4.5.4	Identification de la route	138
4.5.5	Application aux données réelles	141
4.6	Conclusion	151
Conclusion générale		155
Liste des tableaux		159
Table des figures		161
Bibliographie		165

Glossaire

Map matching

$\overline{\mathcal{N}}$: ensemble prédéfini de routes
\mathcal{N}	: ensemble de courbes représentant les routes
P^t	: position du véhicule à l'instant t
$\Delta\beta^t$: écart entre la direction du véhicule et celle de la route ramené à l'intervalle $[-180^\circ, 180^\circ]$
α_i	: angle de position relative de la route i
WS_H	: score de pondération de la similarité cap du véhicule / direction de la route
WS_{PD}	: score de pondération de la distance
WS_{RP}	: score de pondération de la position relative
TWS	: score de pondération total d'un segment
σ_x^2	: variance de l'erreur sur la position du véhicule sur l'axe x
σ_y^2	: variance de l'erreur sur la position du véhicule sur l'axe y
σ_{xy}	: covariance des erreurs sur la position du véhicule sur les axes x et y
P^S	: position GPS
P^{gps}	: projection perpendiculaire de la position GPS sur le segment candidat
P^{map}	: position calculée à l'aide de la carte numérique et la vitesse
P^{mm}	: position map matchée
$\sigma_{map,x}^2$: variance de l'erreur sur la position trouvée à l'aide de la carte numérique et la vitesse sur l'axe x
$\sigma_{map,y}^2$: variance de l'erreur sur la position trouvée à l'aide de la carte numérique et la vitesse sur l'axe y
$\sigma_{gps,x}^2$: variance de l'erreur sur la position GPS sur l'axe x
$\sigma_{gps,y}^2$: variance de l'erreur sur la position GPS sur l'axe y
\hat{x}	: estimée de la position du véhicule sur l'axe x
\hat{y}	: estimée de la position du véhicule sur l'axe y

Estimation

x_k	: vecteur d'état à l'instant k
z_k	: vecteur des observations à l'instant k
Z_k	: observations jusqu'à l'instant k
F	: matrice de transition
H	: matrice d'observation
G	: matrice de commande
Q	: matrice de covariance du bruit du processus
R	: matrice de covariance du bruit des mesures
\hat{x}_k	: estimée à l'instant k
\tilde{x}_k	: erreur d'estimation à l'instant k
\hat{x}^{ML}	: estimateur à maximum de vraisemblance
\hat{x}^{MAP}	: estimateur du maximum a posteriori
\hat{x}^{LS}	: estimateur des moindres carrés
\hat{x}^{MMSE}	: estimateur au sens des moindres carrés
$\mathcal{N}(x \mu, \sigma^2)$: indique que x suit une loi normale de moyenne μ et de variance σ^2
Λ	: fonction de vraisemblance
$\hat{x}_{k+1 k}$: prédiction du filtre à l'instant $k+1$
$P_{k+1 k}$: matrice de covariance de la prédiction à l'instant $k+1$
$\hat{x}_{k+1 k+1}$: moyenne de la loi <i>a posteriori</i> du filtre à l'instant $k+1$
P_{k+1}	: matrice de covariance de la loi <i>a posteriori</i> du filtre à l'instant $k+1$
K_k	: gain du filtre de Kalman

Estimation multi-modèles

M^j	: modèle j
p_{ij}	: probabilité de transition
μ_k^j	: probabilité du modèle j à l'instant k
$\mu_k^{i j}$: probabilité de mixage à l'instant k
Λ_k^j	: fonction de vraisemblance du filtre j à l'instant k
ν_k^j	: innovation du filtre j à l'instant k
S_k^j	: matrice de covariance de l'innovation du filtre j à l'instant k
$\hat{x}_{k-1 k-1}^{0j}$: Initialisation du filtre j à l'instant k
$P_{k-1 k-1}^{0j}$: matrice de covariance de $\hat{x}_{k-1 k-1}^{0j}$
$\hat{x}_{k k}^j$: estimée fournie par le filtre j à l'instant k
$P_{k k}^j$: matrice de covariance de $\hat{x}_{k k}^j$
$\hat{x}_{k k}$: estimée du filtre multi-modèles à l'instant k

$P_{k|k}$: matrice de covariance de $\hat{x}_{k|k}$

Estimation multi-modèles circulaire

$\phi_{f,k}$: mesure angulaire fusionnée
 $\kappa_{f,k}$: paramètre de concentration du résultat de la fusion
 $\hat{\theta}_k^{0j}$: initialisation du filtre circulaire j à l'instant k
 $P_{\theta_k}^{0j}$: paramètre de concentration de $\hat{\theta}_k^{0j}$
 $v_{\theta_k}^j$: vraisemblance du modèle j à l'instant k
 $S_{v_{\theta_k}}^j$: paramètre de concentration de $v_{\theta_k}^j$ à l'instant k
 $\hat{\theta}_k^j$: angle estimé par le filtre circulaire j à l'instant k
 $P_{\theta_k}^j$: paramètre de concentration de $\hat{\theta}_k^j$ à l'instant k
 $\hat{\theta}_k$: angle estimé par le filtre CIMM à l'instant k
 P_{θ_k} : paramètre de concentration de $\hat{\theta}_k$ à l'instant k

Map matching particulière circulaire

θ : angle de direction du véhicule
 ϕ : mesure d'angle de direction affectée de bruit
 v : vitesse du véhicule
 Δt : pas de calcul
 x, y : position du véhicule sur un plan 2D
 $\tilde{\theta}$: variation de l'angle θ pendant Δt
 $r(t)$: identité de la route courante à l'instant t
 $\alpha(r(t))$: direction de la route $r(t)$
 X_k^i, Y_k^i : coordonnées de la particule i à l'instant k
 R_k^i : segment de route sur lequel se trouve la particule i à l'instant k
 \tilde{w}_k^i : poids de la particule i à l'instant k
 w_k^i : poids normalisé de la particule i à l'instant k
 \hat{x}_k, \hat{y}_k : coordonnées de la position estimée du véhicule à l'instant k
 f_{CN} : densité de probabilité de la distribution de von Mises
 κ : paramètre de concentration d'une distribution de von Mises
 $\mathcal{CN}(\theta|\mu, \kappa)$: indique que θ suit une distribution de von Mises de moyenne μ et de paramètre de concentration κ
 $\mathcal{WN}(\theta|\mu, \rho)$: indique que θ suit une distribution Wrapped Normal de moyenne μ et de variance $\sigma^2 = -2\log(\rho)$
 A : fonction d'approximation, ratio de I_1 et I_0
 A^{-1} : fonction inverse de la fonction A

I_0	: fonction de Bessel modifiée de la première espèce, d'ordre 0
I_1	: fonction de Bessel modifiée de la première espèce, d'ordre 1
$\hat{\theta}_{k k-1}$: angle prédit par le filtre
$P_{\theta_{k k-1}}$: paramètre de concentration de l'angle prédit
$\hat{\theta}_k$: angle estimé par le filtre
P_{θ_k}	: paramètre de concentration de l'angle estimé
$f(\cdot), f(\cdot \cdot)$: densités de probabilité
S	: somme des sinus d'un vecteur d'angles
C	: somme des cosinus d'un vecteur d'angles
\arctan^*	: fonction arctangente "quadrant-specific"
Θ	: vecteur d'angles à estimer
$\tilde{\Theta}$: vecteur des variations angulaires à estimer
κ_R	: paramètre de concentration du bruit de mesure
κ_Q	: paramètre de concentration du bruit de modèle sur l'angle
$\kappa_{\tilde{Q}}$: paramètre de concentration du bruit de modèle sur la variation angulaire
κ_{Rp}	: paramètre de concentration de la distribution de von Mises utilisée dans l'affectation des poids aux particules
dc	: distance circulaire
$\Delta\alpha$: angle entre les routes de la bifurcation en Y

Abréviations

AMM	: Autonomus Multiple Model
BMM	: B-Best Multiple Model
CMM	: Cooperating Multiple Model
CIMM	: IMM circulaire
DR	: Dead Reckoning
EKF	: Extended Kalman Filter
FSMM	: Fixed Structure Multiple Model
GNSS	: Global Navigation Satellite System
GPB	: Generalized Pseudo-Bayesian
GPB1	: First Order Generalized Pseudo-Bayesian
GPB2	: Second Order Generalized Pseudo-Bayesian
GPRS	: General Packet Radio Service
GPS	: Global Positioning System
IMM	: Interacting Multiple Model
KF	: Kalman Filter
LS	: Least squares
MAP	: Maximum A Posteriori
MAP-TOP	: Map matching topologique
MAP-SIR	: Map matching circulaire particulière
MAP-SIR-CIR	: Map matching circulaire particulière avec préfiltrage circulaire
MAP-SIR-CIMM	: Map matching circulaire particulière avec préfiltrage CIMM
ML	: Maximum Likelihood
MM	: Multiple Model
MMSE	: Minimum Mean Square Error
RIMM	: Reweighted Interacting Multiple Model
RAMS	: Recursive Adaptive Model-Set
SIR	: Sequential Importance Resampling
SLAM	: Simultaneous Location And Mapping

Abréviations

VSMM	:	Variable Structure Multiple Model
VMM	:	Viterbi Multiple Model

Introduction générale

Beaucoup d'applications industrielles, militaires ou de recherche nécessitent une localisation précise des objets mobiles. Les systèmes de navigation satellitaires, et notamment le GPS, permettent un positionnement absolu si un minimum de 4 satellites est visible. Toutefois, plusieurs contraintes diminuent la précision de ce type de positionnement dans le milieu urbain. On cite particulièrement le masquage des signaux satellitaires par les constructions élevées, les multi-trajets et la dilution de précision due aux positions relatives des satellites visibles. Dans des circonstances extrêmes, les signaux des satellites sont complètement masqués et le positionnement absolu n'est plus possible. L'une des solutions mises en œuvre pour corriger la position dans de telles situations est la fusion des données GPS avec celles collectées des capteurs proprioceptifs (odomètre, gyroscope, magnétomètre, ...). Dans le cas de masquage on a recours temporairement à la navigation par l'estime (*DR : Dead Reckoning*).

La technique qui permet d'améliorer la localisation est le Map matching. Dans cette approche, la position obtenue par GPS/DR est mise en correspondance avec une carte vectorielle en vue de trouver le segment de route qui lui correspond. Cette technique améliore la précision de la position par l'ajout de contraintes supplémentaires au processus de localisation comme la géométrie ou la topologie du réseau de routes. Dans la plupart des algorithmes proposés dans la littérature, le processus de map matching est une étape postérieure au calcul de la position. L'objectif consiste alors à trouver la route qui respecte des critères. Ces critères peuvent être par exemple, la proximité de la route par rapport à la position mesurée, ou l'écart minimal entre la direction de la route et celle du véhicule. Toutefois les performances de l'algorithme de Map matching restent très dépendantes de la précision des capteurs absolus et proprioceptifs. Dans certains cas, la position déduite par Map matching est incorrecte car elle repose sur une position erronée à l'origine.

Le sujet de recherche développé dans cette thèse traite le problème de Map matching dans les environnements contraints, mais connus. On peut citer les ports, les zones industrielles, ou les milieux urbains denses pour lesquels la visibilité des satellites GPS n'est pas garantie, mais

pour lesquels on dispose d'une connaissance supposée parfaite du réseau routier. Des capteurs de vitesse et de direction sont utilisés pour localiser le véhicule pendant les masquages des signaux satellitaires. L'idée qui est développée consiste à utiliser, en cas d'indisponibilité du capteur GPS, les informations absolues de direction fournies par un magnétomètre pour réaliser le Map matching. La mise en correspondance des mesures avec la carte ne se fait donc plus à partir de la position sur un segment de route, mais à partir de la direction du segment de route.

Plusieurs techniques sont proposées pour améliorer la précision de la localisation. D'une part, les mesures de direction sont préfiltrées par filtrage circulaire. Le choix de ce filtre est motivé par la nature circulaire des observations dont le traitement fait appel à des statistiques différentes de celles du domaine linéaire. D'autre part, l'algorithme de Map matching utilise un filtre particulière défini aussi dans le domaine circulaire, conjointement avec une carte numérique. L'originalité de cet algorithme réside dans l'utilisation des particules pour l'exploration de l'espace des routes probables. Ces particules sont propagées selon la vitesse du véhicule sur différentes routes du réseau tant que leur probabilité est toujours significative. Ces probabilités sont définies à partir de la distribution statistique circulaire de von Mises qui reflète l'écart entre la direction du véhicule et la direction de la route sur laquelle évolue la particule. On note ici que les virages jouent un rôle crucial dans cette approche, car ils permettent d'éliminer les particules improbables qui ne sont pas en accord avec la manœuvre observée.

Dans ce contexte, on montre que la détection des manœuvres du véhicule, à partir des mesures de sa direction, permet d'améliorer le Map matching. La méthode proposée dans ce travail de recherche repose sur la mise en œuvre de l'approche multi-modèles à partir des observations de directions. L'algorithme multi-modèles proposé est défini dans le domaine circulaire et vise un double objectif. D'une part, améliorer l'estimation de la direction par le changement de modèles pendant les manœuvres, et d'autre part d'évaluer le comportement du véhicule sous une forme probabiliste. Cela constitue une autre originalité de ce travail, car le comportement du véhicule ainsi défini sera utilisé pour orienter les particules vers les routes les plus probables et sera utilisé pour améliorer ainsi la précision du Map matching.

Ce document de thèse est composé de quatre chapitres. Les deux premiers chapitres bibliographiques font un état de l'art des techniques qui seront le fondement des développements proposés dans les deux chapitres suivants. Les chapitres trois et quatre présentent les méthodes proposées et leur évaluation sur données synthétiques. C'est dans le quatrième et dernier chapitre que la faisabilité sur données réelles est montrée.

Dans le premier chapitre nous réalisons l'état de l'art des méthodes de Map matching. On y expose les différentes approches existantes et les limitations de chacune d'entre elles. On détaille plus particulièrement l'approche de Map matching topologique que nous avons choisi comme approche de référence dans l'évaluation de nos travaux. Les raisons de ce choix sont que cet algorithme utilise un processus de décision multi-critères qui prend en compte la topologie du réseau routier. Cela le rend plus robuste dans les cas où les capteurs sont défaillants. Cet algorithme proposé par Quddus est l'algorithme de référence de la littérature.

Le second chapitre traite du filtrage multi-modèles. On aborde dans sa première partie la théorie de l'estimation et du filtrage d'état en général. On y décrit le fonctionnement de deux filtres de référence qui sont le filtre de Kalman et le filtre particulaire. Dans la deuxième partie de ce chapitre, on passe en revue les différents estimateurs multi-modèles existants avec une description plus détaillée du filtre IMM utilisé dans notre approche. En effet, c'est dans le chapitre quatre qu'une version circulaire de ce filtre est proposée.

Dans le troisième chapitre, on expose le principe de la méthode de Map matching particulière proposée avec une étude de plusieurs variantes de préfiltrage de la direction. La seconde partie de ce chapitre est expérimentale. On y évalue les performances de notre algorithme de Map matching dans deux cas synthétiques : le cas problématique d'une jonction sous forme d'Y, et le cas du contexte réel avec une trajectoire reconstruite à partir de données réelles. On compare notre méthode avec l'algorithme topologique classique dans des cas de masquage aléatoire des signaux GPS.

Le quatrième chapitre traite l'amélioration apportée au map matching proposé par l'approche multi-modèles. Il débute par une description théorique du filtre IMM circulaire que nous proposons (noté CIMM), et est suivi par la description de sa mise en œuvre pour le Map matching particulaire. Le filtre proposé est évalué pour les expérimentations de la bifurcation en Y et celle du contexte réel. Dans la dernière partie de ce chapitre, les algorithmes étudiés seront évalués sur des données réelles afin de montrer leurs performances mais aussi que leurs limitations.

Enfin, une conclusion générale fera le bilan des travaux réalisés dans cette thèse et donnera une vue sur les perspectives futures.

Chapitre 1

Map matching

1.1 Introduction

L'utilisation croissante du positionnement par GPS a aidé durant les dernières décennies à la localisation des objets mobiles et des personnes. Cependant, à cause des erreurs inhérentes au système GPS, la déviation de la trajectoire fournie par le capteur GPS de la trajectoire réelle est inévitable. Pour y remédier, on cherche à associer les positions GPS mesurées aux routes réelles sur une carte digitale. Ce processus est communément appelé *Map matching*.

L'objectif général du Map matching consiste à identifier la route réelle sur laquelle se déplace la personne ou le véhicule. C'est une procédure clé dans beaucoup d'applications reposant sur la géo-localisation comme : la gestion de flottes et les systèmes de transports intelligents. On ne considère pas ici les approches développées dans le cas de la problématique SLAM (*Simultaneous Location And Mapping*) dont l'objectif est d'identifier les paramètres des routes, ou bien de reconstruire la carte, et de géo-localiser le véhicule.

Plusieurs algorithmes de Map matching ont été développés au fil des dernières années. Ils sont classés selon plusieurs catégories. La première prend en considération les relations géométriques entre le point GPS et la carte, la seconde considère en plus la topologie du réseau routier et l'historique des données GPS. Généralement cette seconde catégorie donne de meilleurs résultats [89]. D'autres algorithmes utilisent l'approche statistique, la logique floue ou d'autres méthodes avancées.

1.2 Position du problème

Avant d'étudier les différents algorithmes existant dans la littérature, nous devons définir les divers éléments d'un réseau routier sur une carte numérique.

Comme décrit dans [88], un véhicule se déplace en général sur un ensemble prédéfini de routes, noté $\overline{\mathcal{N}}$ (Figure 1.1a). Nous n'avons pas une connaissance précise de ce réseau, nous disposons uniquement d'une représentation du réseau notée \mathcal{N} qui consiste en un ensemble de courbes sur \mathbb{R}^2 , chaque courbe étant appelée un *arc* ou *lien* (Figure 1.1b). Chaque arc est supposé constitué d'un ensemble de *segments* linéaires. Ainsi, un arc $A \in \mathcal{N}$ peut être complètement défini par une séquence finie de points $(A^0, A^1, \dots, A^{n_A})$. Les points extrêmes A^0 et A^{n_A} sont appelés des *nœuds*, alors que les points $(A^1, A^2, \dots, A^{n_A-1})$ sont dits *sommets* (*vertex*). Un nœud est le point du début/fin d'un arc, ou encore un point de transition d'un arc à un autre (carrefour, intersection entre routes, ...).

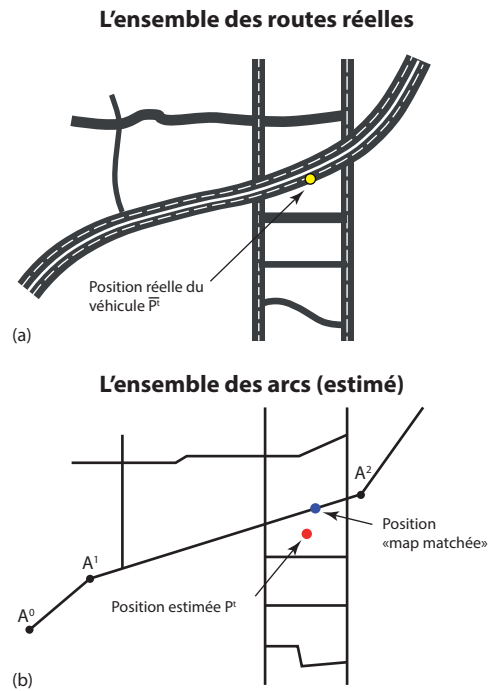


FIGURE 1.1 – Problème du Map matching

Lorsqu'un véhicule se déplace sur ce réseau de routes, alors nous disposons, pour un nombre fini d'instants $\{0, 1, \dots, T\}$, des positions estimées du véhicule issues des capteurs utilisés. Une position estimée à l'instant t est notée P^t alors que la position réelle du véhicule au même instant est notée \bar{P}^t . L'objectif d'un problème de Map matching est de repérer ou "*matcher*"

la position estimée P^t sur un arc A de la carte \mathcal{N} , puis de déterminer la route $\bar{A} \in \bar{\mathcal{N}}$ qui correspond à la position réelle du véhicule \bar{P}^t .

1.3 Analyse géométrique

Ces algorithmes de Map matching se basent sur la relation géométrique entre le point GPS et les éléments de la carte numérique. Ils ne prennent pas en compte la façon avec laquelle les arcs sont reliés. Plusieurs critères sont considérés [66] :

- Distance Point-à-Point (*point-to-point distance*)
- Distance Point-à-Courbe (*point-to-curve distance*)
- Distance Courbe-à-Courbe (*curve-to-curve distance*)
- Angle Courbe-à-Courbe (*angle of curve-to-curve*)

1.3.1 Map matching « point-to-point »

L'algorithme le plus trivial dans l'analyse géométrique et connu sous le nom *point-to-point matching* [16]. C'est un simple algorithme de recherche qui repose sur le dépistage, pour chaque point, du nœud ou du sommet le plus proche. Plusieurs algorithmes différents existent dans la littérature pour faire cette sélection, comme par exemple [13] et [28]. Les avantages de cette approche sont la facilité d'implémentation et la rapidité d'exécution. Cependant, cette approche est très sensible à la manière avec laquelle la carte est digitalisée. Ainsi, les arcs ayant plus de sommets ont plus de probabilité d'être choisis, tandis que dans le cas d'un long arc linéaire ayant seulement deux nœuds aux extrémités, toutes les positions le long de cet arc seront *matchées* (rapprochées, rattachées, accolées) à ces deux nœuds. La figure 1.2 illustre cette approche.

Sur cette figure, les positions issues du système de navigation sont indiquées par les points P^1 à P^7 pour un véhicule naviguant dans un réseau de dix nœuds (A à J). Le véhicule traverse en réalité le lien AB puis BD. Néanmoins, les résultats de l'algorithme *point-to-point* montrent que le véhicule emprunte les liens AB, BE, EH et finalement BD, ce qui est incorrect.

1.3.2 Map matching « point-to-curve »

Une autre approche géométrique de Map matching est le *point-to-curve matching* ([16] et [88]). La position obtenue du système de navigation est *matchée* à l'arc le plus proche du ré-

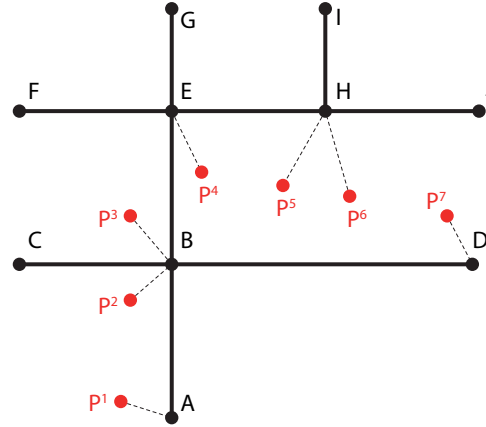


FIGURE 1.2 – Matching point-to-point

seau. Chaque arc étant constitué de segments linéaires, on calcule la distance de la position mesurée à chaque segment linéaire. Le segment pour lequel la distance calculée est minimale est sélectionné.

Selon White [88], le calcul de la distance entre un point et un segment n'est pas une simple projection perpendiculaire comme dans le cas d'une ligne, mais elle doit prendre en considération d'autres conditions. Sa méthode est présentée sur la figure 1.3 : Le calcul de distance entre le point p et le segment $[A^0 A^1]$ dans (a) est direct puisque le point projeté perpendiculairement appartient au segment. En revanche, dans (b), la projection perpendiculaire du point q se trouve en dehors du segment $[A^0 A^1]$. Dans ce cas, la distance considérée est le minimum des deux distances $\|qA^0\|$ et $\|qA^1\|$. De plus, un arc étant constitué de plusieurs segments linéaires, la distance entre un point et un arc est le minimum des distances calculées entre le point d'intérêt et les différents segments constituant l'arc.

Bien que l'approche *point-to-curve* donne des résultats meilleurs que l'approche *point-to-point*, elle souffre de plusieurs défauts qui la rendent souvent inadéquate en pratique. En effet, le segment le plus proche n'est pas toujours le segment correct. Sur la figure 1.4, par exemple, un véhicule se déplaçant le long du segment A peut voir sa position P^1 matchée par erreur au segment B car la distance de P^1 à ce segment est la plus petite.

En reprenant l'exemple de la figure 1.2 et en appliquant l'algorithme du matching *point-to-curve*, on trouve les résultats de la figure 1.5. On constate que l'algorithme sélectionne les segments AB, BE puis BD, alors que le véhicule emprunte en réalité AB puis BD. Les résultats de cette méthode sont donc instables dans un réseau urbain dense [70].

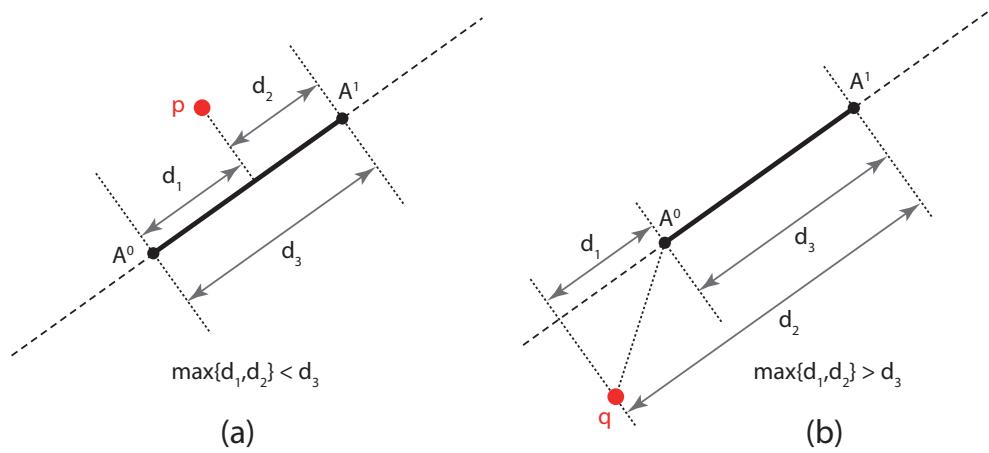


FIGURE 1.3 – Calcul de distance entre un point et un segment

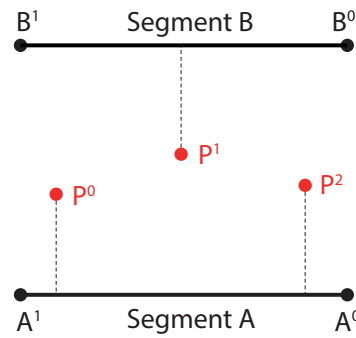


FIGURE 1.4 – Erreur sur le matching point-to-curve

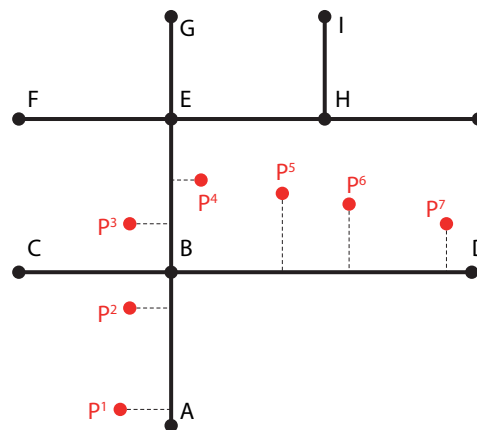


FIGURE 1.5 – Matching point-to-curve

Srinivasan et al. proposent dans [74] une amélioration de l'algorithme du matching *point-to-curve*. Leur étude recommande de faire un double test : test de direction et test de rotation

avant de matcher la position mesurée à l'arc le plus proche de la carte. Le test de direction est effectué entre la direction du segment de la route et le cap instantané du véhicule en provenance du capteur GPS. Le test de rotation est introduit pour s'assurer que la différence de direction entre deux mesures GPS consécutives corresponde à la rotation calculée à partir de la topologie du réseau de routes. Ils confirment dans leur étude que leur algorithme avancé de matching *point-to-curve* améliore l'identification du lien correct. Cependant l'effet de la vitesse du véhicule sur la direction mesurée est ignoré dans cette étude. En effet, le cap issu d'un capteur GPS devient imprécis pour de faibles vitesses [83]. De plus, cette méthode est sensible aux données aberrantes (*outliers*) [70].

1.3.3 Map matching « curve-to-curve »

D'autres algorithmes utilisant la distance courbe-à-courbe (*curve-to-curve*) ont été développés [15, 88, 67]. Cette approche considère simultanément m positions du véhicule (P^1, P^2, \dots, P^m) et en construit une courbe de segments linéaires, puis cherche sur la carte numérique la courbe qui lui est proche en termes de distance. Il existe plusieurs définitions de la distance entre deux courbes A et B sur \mathbb{R}^2 . Une définition triviale est celle qui repose sur le minimum de distance entre les deux courbes :

$$\|A - B\|_{min} = \min_{a \in A, b \in B} \|a - b\| \quad (1.1)$$

Bien que cette distance soit suffisante dans plusieurs circonstances, elle ne fonctionne pas correctement en Map matching et elle est assez sensible aux données aberrantes [15]. Ceci est illustré sur la figure 1.6 qui montre qu'avec cette définition de distance la courbe P sera plus proche de A que de B.

Une autre mesure de distance est proposée dans [15] par Bernstein et Kornhauser. Elle consiste à définir les courbes A, B et P comme étant des courbes paramétrées et consiste à calculer par une intégrale sur le temps, la distance entre chaque point d'une courbe et le point correspondant sur l'autre courbe. Cette méthode améliore la précision du calcul de la distance mais elle peut donner des résultats inattendus tel qu'illustré sur la figure 1.7. Sur cette figure, la courbe formée par la trajectoire du véhicule (points P^1 à P^7) est plus proche de la courbe B qui lui est perpendiculaire qu'à la courbe A. On serait donc amené à reporter la trajectoire du véhicule à la courbe B plutôt qu'à A, ce qui est incorrect.

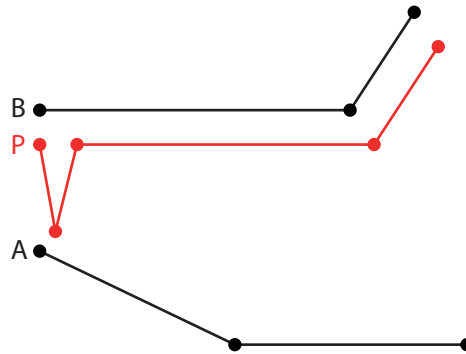


FIGURE 1.6 – Problème de la distance entre deux courbes en matching curve-to-curve

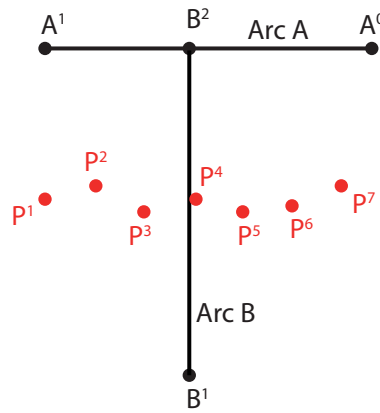


FIGURE 1.7 – Défaut du matching curve-to-curve

Une amélioration de ce même algorithme a été proposée par Bernstein et Kornhauser [15]. Leur approche introduit des informations topologiques. Elle commence par identifier le nœud candidat en utilisant le matching *point-to-point*. Ensuite, elle construit une courbe de segments linéaires à partir de chaque chemin partant de ce nœud (voir Figure 1.8). Dans une seconde phase, elle construit une courbe de segments linéaires à partir de la trajectoire du véhicule, puis calcule la distance entre cette courbe et les courbes déterminées précédemment sur la carte. L'arc de route le plus proche à la courbe de la trajectoire est sélectionné comme étant la route que le véhicule aurait probablement emprunté.

Notons que dans l'algorithme Bernstein et Kornhauser, et dans le contexte du Map matching *curve-to-curve* d'une manière générale, on est amené à calculer la distance entre deux courbes de longueur différente comme le montre la figure 1.9. La première étant la trajectoire du véhicule P et la seconde un arc candidat A du réseau des routes. On peut calculer la distance entre les deux courbes en utilisant simplement leurs longueurs réelles (figure 1.9b), dans ce cas la distance est $m + n'$, mais ceci risque de fournir une mesure incorrecte qui s'accroît lorsque la

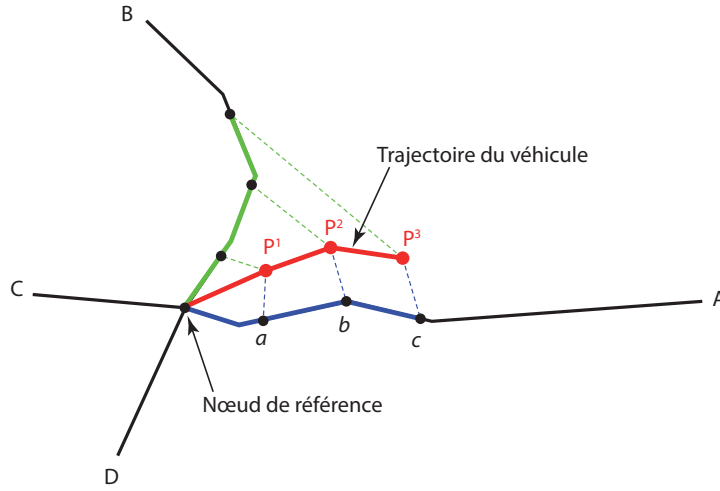


FIGURE 1.8 – Matching curve-to-curve

différence de longueur devient importante (n' devient de plus en plus grand). On peut améliorer cette mesure en considérant seulement une portion de longueur égale à la courbe courte P sur la courbe longue A (figure 1.9a), et dans ce cas la distance devient $m + n$, quelle que soit la différence de longueur.

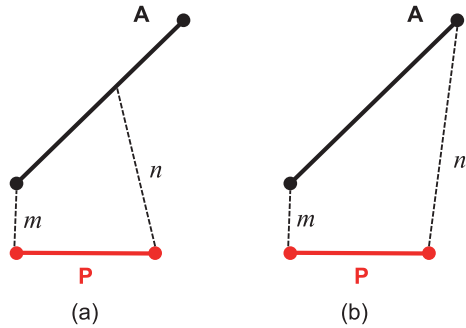


FIGURE 1.9 – Distance entre courbes de longueur différente

En revenant à l'algorithme de Bernstein et Kornhauser sur la figure 1.8. La projection des points de la trajectoire P^1 , P^2 et P^3 sur l'arc A correspond aux points a , b et c respectivement de façon à ce que la distance entre P^0 et P^1 soit égale à celle entre P^0 et a en parcourant l'arc A , et la distance entre P^1 et P^2 soit égale à la distance entre a et b et ainsi de suite. La distance entre la trajectoire et l'arc A est la somme des distances $\|P^1a\|$, $\|P^2b\|$ et $\|P^3c\|$. On procède de la même manière pour l'arc B .

Finalement, l'algorithme de Bernstein et Kornhauser est sensible aux données aberrantes et dépend, pendant sa phase d'initialisation, du matching *point-to-point*, ce qui peut donner parfois

des résultats inattendus [15].

1.3.4 Matching de surface (« *Area matching* »)

Lianxia dans [57] présente un algorithme utilisant la cartographie, il s'appuie sur le calcul de la surface entre le lien et la trajectoire du véhicule pour positionner celle-ci sur la carte numérique.

La distance calculée à partir de la surface entre deux lignes est définie par :

$$d = \frac{S}{L} \quad (1.2)$$

Où S est la surface du polygone formé par les deux lignes et L la longueur moyenne des deux lignes. La distance d est une mesure de proximité entre les deux lignes, elle augmente proportionnellement avec la surface S qui grandit lorsque les deux lignes s'éloignent.

Si les deux lignes ont une intersection comme sur la figure 1.10, on sépare les polygones concaves et convexes en considérant leur surfaces positive dans les deux cas. S est la somme de toutes les surfaces des polygones engendrés (surface grisée sur la figure 1.10).

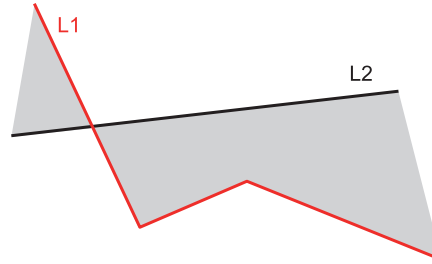


FIGURE 1.10 – Surface entourée par deux lignes

Pour chaque polygone ayant N sommets, la surface est donnée par la formule suivante où (x_i, y_i) sont les coordonnées du sommet i :

$$S = \left| \frac{1}{2} \sum_{i=1}^{N-2} (x_i y_{i+1} - x_{i+1} y_i) + (x_N y_1 - x_1 y_N) \right| \quad (1.3)$$

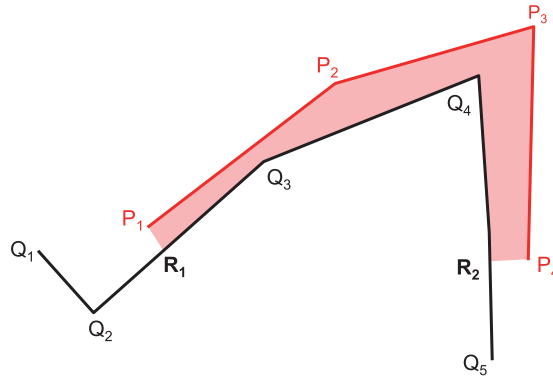


FIGURE 1.11 – Méthode "Area matching"

Sur la figure 1.11 le principe de la méthode *Area matching* est présentée. La méthode consiste à calculer la surface entre le lien formé par les points Q_1 à Q_5 et la trajectoire du véhicule formée par les points P_1 à P_4 . La méthode commence par projeter les extrémités de la trajectoire P_1 et P_4 sur le lien, donnant naissance aux points R_1 et R_2 respectivement. Ensuite, on calcule la surface entre les deux courbes (P_1, P_4) et (R_1, R_2) par la formule (1.3) (surface ombrée sur la figure 1.11 qui correspond au polygone $(P_1, P_2, P_3, P_4, R_2, Q_4, Q_3, R_1)$), puis on détermine la distance par la formule (1.2). Sur tous les liens candidats, celui ayant le minimum de distance est le lien sélectionné.

On voit clairement que la longueur de la trajectoire du véhicule (P_1, P_4) devient plus courte après projection sur le lien (courbe (R_1, R_2)) ce qui introduit des erreurs. Lianxia [57] indique que la précision de cet algorithme varie entre 79.5% et 90% selon la complexité du réseau routier et qu'il n'est pas adapté à une utilisation en temps réel à cause du temps de calcul important requis par le calcul des surfaces surtout dans des réseaux urbains denses. Il propose l'amélioration de cet algorithme par la prise en considération de la direction des routes.

1.4 Analyse topologique

Dans les systèmes d'informations géographiques, la Topologie fait allusion à la relation entre les entités tel que les points, les lignes et les polygones. Cette relation peut être une relation d'adjacence (pour les polygones), connectivité (pour les lignes) ou appartenance (pour les points et les polygones). On ne doit pas confondre la topologie à la topographie qui s'intéresse à la nature et à l'altitude des surfaces (montagnes, vallées, ...). Un algorithme de Map matching qui utilise la géométrie des liens aussi bien que la connectivité et la contiguïté des liens est connu sous le nom d'*algorithme de Map matching topologique* [29, 24, 64].

1.4.1 Algorithme topologique

Dans [29], Greenfeld propose un algorithme topologique utilisant la pondération. Son algorithme repose uniquement sur les positions observées de l'objet mobile sans prendre en considération ni la direction, ni la vitesse issues du capteur GPS. Son algorithme se déroule en deux sous-algorithmes : 1) *InitialMapping* qui cherche à localiser la première position sur la carte et 2) *Map* qui recherche les positions suivantes sur le lien repéré.

Le sous-algorithme *InitialMapping* cherche le nœud le plus proche à la première position issue du GPS, puis détermine tous les liens qui y sont connectés. Lorsque la position suivante est disponible, elle est rattachée à l'un de ces liens. Ce sous-algorithme est appliqué dans trois cas :

- Pour la première position reçue du capteur GPS
- Lorsque la distance entre deux positions GPS successives dépasse un seuil de tolérance, par exemple lorsque le récepteur GPS ne fournit plus de mesures à cause d'un masquage de satellites
- Lorsque le sous-algorithme *Map* n'est pas capable de sélectionner un lien pour la position GPS

Les sous-algorithme *Map* repose sur la première position calculée par *InitialMapping* pour faire une analyse topologique, utilisant un système de pondération. Celui-ci repose sur deux critères qui sont la proximité et l'orientation. Deux mesures sont utilisées pour évaluer la proximité :

- La première est la distance perpendiculaire entre la position mesurée et le lien. Cette mesure géométrique n'est pas toujours suffisante comme discuté dans le Map matching "*point-to-curve*"
- Pour la deuxième mesure, on cherche s'il existe une intersection de la ligne formée par le point GPS précédent et l'actuel avec le segment candidat. Sur la figure 1.12, si le point actuel est P^2 , l'intersection de P^1P^2 avec le segment A est définie par le point a . Ce critère n'est valide que si l'intersection se produit entre les deux points P^1 et P^2 et le segment A , et non pas sur leur prolongation (comme dans l'exemple du point d). De plus, il faut que l'angle entre les deux lignes en intersection soit faible. Par conséquent, le facteur de pondération associé à cette mesure dépend du cosinus de l'angle d'intersection.

Le critère d'orientation considère l'écart entre l'azimut du véhicule et celui du lien ; l'azimut du véhicule étant défini par la ligne entre le point GPS actuel et précédent. L'utilisation des azimuts est justifiée par le fait que les angles sont mesurés par rapport au Nord. Il est ainsi possible de différencier entre deux segments parallèles mais de directions opposées. Le facteur de pondération utilise la fonction cosinus pour favoriser les faibles écarts d'azimut.

La pondération totale d'un lien est la somme de trois facteurs de pondération comme indiqué par la formule (1.4).

$$W = W_D + W_I + W_{AZ} \quad (1.4)$$

avec :

- W : Score total du lien
- W_D : Pondération de proximité
- W_I : Pondération d'intersection, si elle existe
- W_{AZ} : Pondération de similarité d'orientation

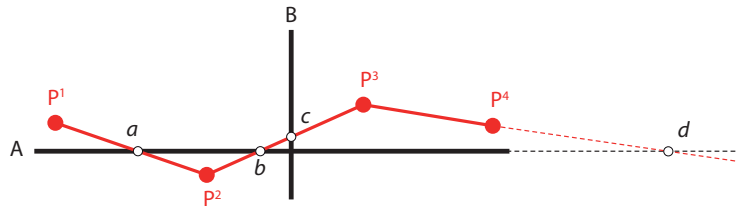


FIGURE 1.12 – Algorithme de Greenfeld : Intersections entre ligne GPS et liens

Après identification du lien correct en utilisant l'un des algorithmes *InitialMapping* ou *Map*, la position *matchée* est trouvée par projection orthogonale du point GPS sur le lien.

Cette méthode est sensible aux données aberrantes [70], en plus, elle ne prend pas en compte la vitesse du véhicule. En effet, pour des vitesses inférieures à 3 m/s les positions GPS sont distribuées aléatoirement au voisinage de la position réelle [83]. Un autre défaut de cette méthode montre que lorsque le véhicule se déplace le long d'un lien, la pondération d'intersection est nulle car la trajectoire, étant parallèle au lien, n'a aucun point d'intersection avec celui-ci, ce qui diminue le score du lien, bien qu'il soit le lien correct.

Un autre défaut de cette méthode est montré au niveau des jonctions tel qu'illustré par la figure 1.13. Sur cette figure, le véhicule se déplace le long de AC. Pour les positions P^1 et P^2 ,

l'algorithme choisit correctement le lien AB. En revanche, pour la position P^3 , l'algorithme identifie le lien BD comme étant le lien correct, car d'une part c'est le lien le plus proche de P^3 , et d'autre part, à l'opposé de BC, il y a une intersection entre P^2P^3 et BD, ce qui confère à BD un score plus grand.

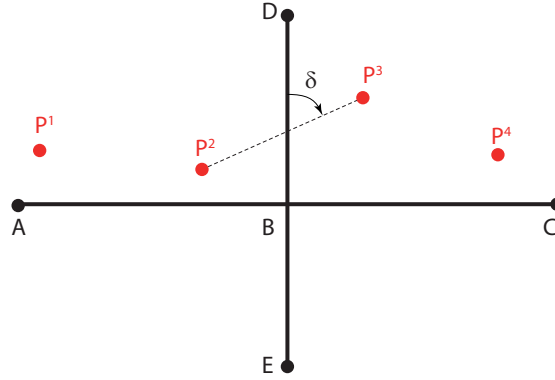


FIGURE 1.13 – Algorithme de Greenfeld : Sélection d'un lien incorrect sur une jonction

Li et Fu ont proposé dans [42] un algorithme de Map matching amélioré reposant sur l'analyse topologique. A l'instar de Greenfeld [29], cette méthode calcule le cap du véhicule à partir de la ligne formée par deux mesures GPS consécutives pour comparer la similarité entre le cap du véhicule et la direction du lien. Leur algorithme se compose de trois étapes : 1) Mode recherche (*Searching mode*), 2) Mode d'exécution normal (*normal running mode*) et 3) Mode rotation (*turning mode*).

En *mode recherche*, l'approche géométrique *point-to-curve* est utilisée pour choisir le lien correct. Dès que ce lien est identifié, l'algorithme passe en *mode d'exécution normal* où il continue à projeter les positions suivantes du véhicule sur ce lien jusqu'à détection d'une manœuvre de rotation. L'algorithme est alors en *mode rotation* et revient au *mode recherche* pour identifier le nouveau lien sur lequel le véhicule s'est déplacé. Les défauts inhérents au Map matching *point-to-curve* discutés précédemment font que cette méthode n'est pas efficace dans les zones urbaines où le réseau routier est dense. En plus, le critère de détection de la manœuvre de rotation n'est pas clairement présenté dans [42].

1.4.2 Algorithme topologique amélioré

Quddus dans [70] a présenté un algorithme amélioré de Map matching reposant sur l'approche topologique et utilisant un système de pondération à l'instar de Greenfeld [29]. Les entrées de cet algorithme sont la position du véhicule issue d'un capteur GPS/DR (DR : *Dead reckoning* représente les données venant de capteurs proprioceptifs montés sur le véhicule), la

vitesse et le cap du véhicule. L'algorithme général est illustré sur la figure 1.14.

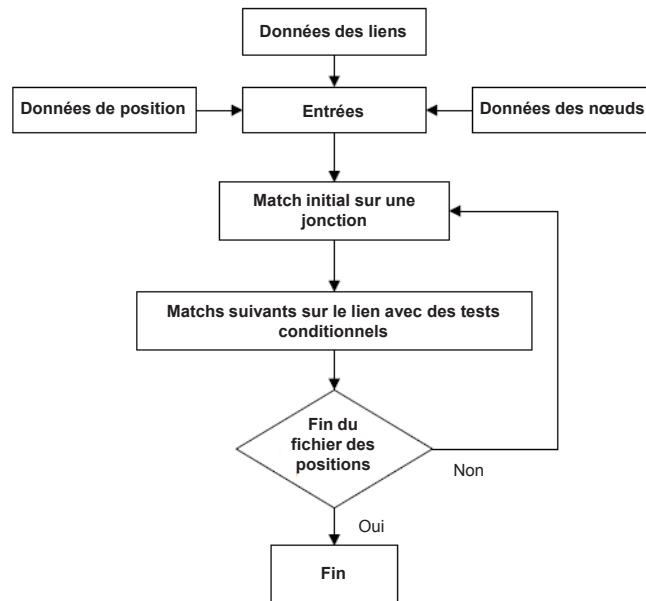


FIGURE 1.14 – Algorithme de Quddus

La difficulté de cette approche réside dans la sélection du lien correct parmi les liens candidats. Cela se produit généralement sur les jonctions, par conséquent un système de pondération est mis en œuvre utilisant plusieurs critères. Quddus a proposé les critères suivants :

- *Cap du véhicule* : calculé à partir des données GPS/DR
- *Proximité* : distance entre la position GPS et le lien
- *Position relative* de la position du véhicule par rapport au lien candidat

Différents facteurs de pondération sont utilisés pour contrôler la contribution de chacun de ces critères dans la prise de décision sur le choix d'un lien candidat. Dans ce qui suit nous allons fournir les détails du calcul de chaque critère.

Pondération du cap du véhicule

Sur la figure 1.15 On suppose que les positions P^1 à P^3 sont correctement placées par l'algorithme de Map matching sur le lien AB. Le point P^4 , en revanche, est problématique car il peut être associé par erreur au lien BC. En considérant la jonction la plus proche B, les segments candidats pour P^4 sont BC, BE et BD. Pour faire le choix entre ces liens un système de pondération reposant sur la comparaison entre le cap du véhicule et la direction du lien peut être utilisé.

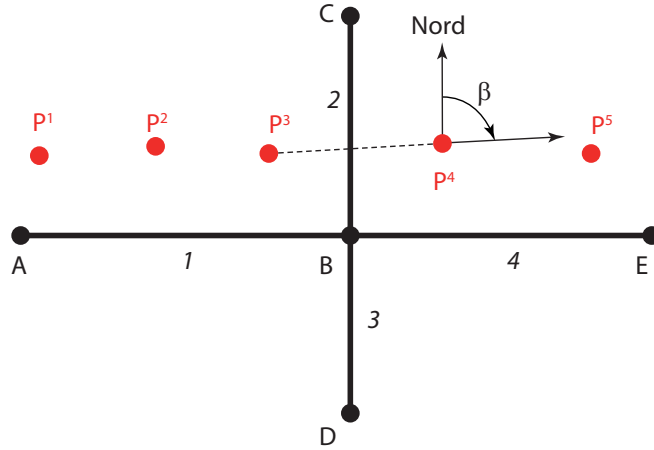


FIGURE 1.15 – Similarité entre le cap du véhicule et la direction du lien

L'angle β représente le cap du véhicule à la position P^4 , c'est l'angle entre le Nord et la ligne formée par P^3 et P^4 . Cet angle donne une bonne indication sur la direction du véhicule en ce point. La direction de chaque lien est notée β' . Elle est déterminée à partir des données de la carte numérique. Dans notre cas $\beta' = 90^\circ$ pour BE, $\beta' = 0^\circ$ pour BC et $\beta' = 180^\circ$ pour BD. La différence entre β et β' est notée $\Delta\beta$. Elle est utilisée pour déterminer la pondération du cap :

$$WS_H = A_H \cdot \cos(\Delta\beta') \quad (1.5)$$

Où :

- WS_H est le score de pondération de la similarité entre le cap du véhicule à la direction du lien,
- A_H est un paramètre positif décrit ci-après par l'équation (1.12). Son rôle est de donner plus ou moins d'importance à WS_H dans la pondération totale d'un lien,
- $\Delta\beta'$ est l'écart entre les angles β et β' ramené à l'intervalle $[-180^\circ, 180^\circ]$. Il est calculé comme suit :

$$\Delta\beta' = \begin{cases} \Delta\beta, & \text{si } -180^\circ \leq \Delta\beta \leq 180^\circ \\ 360^\circ - \Delta\beta, & \text{si } \Delta\beta > 180^\circ \\ 360^\circ + \Delta\beta, & \text{si } \Delta\beta < -180^\circ \end{cases} \quad (1.6)$$

La fonction \cos a été utilisée pour augmenter WS_H lorsque l'écart $\Delta\beta'$ est faible, et pour don-

ner une valeur négative à WS_H pour les liens orientés dans le sens inverse du déplacement du véhicule. De plus, la fonction \cos étant paire, seule l'importance de l'écart $\Delta\beta'$ est considérée, et non son signe.

Pondération de la proximité

La pondération de proximité repose sur la distance obtenue par projection perpendiculaire de la position GPS sur le lien.

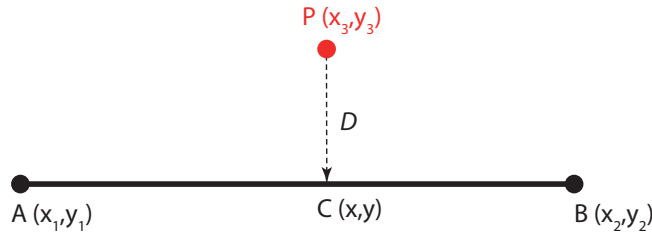


FIGURE 1.16 – Distance perpendiculaire

La distance D est calculée par la formule (1.7) en utilisant les coordonnées des extrémités du segment $A(x_1, y_1)$ et $B(x_2, y_2)$, et le point GPS $P(x_3, y_3)$:

$$D = \frac{x_3(y_1 - y_2) - y_3(x_1 - x_2) + (x_1y_2 - x_2y_1)}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}} \quad (1.7)$$

Lorsque la distance D diminue, le score de pondération doit augmenter. Par conséquent, on définit ce score de la manière suivante :

$$WS_{PD} = A_{PD} \cdot \omega \quad (1.8)$$

Avec :

- WS_{PD} est le score de pondération de la distance D ,
- A_{PD} est le paramètre d'importance de WS_{PD} . Il est défini ci-après par l'équation (1.12),
- ω est un facteur d'échelle variant entre -1 et 1 en fonction de D comme suit :

$$\omega = \begin{cases} 1, & \text{si } D < 5m \\ 1 - 0.01D, & \text{si } 5 \leq D \leq 100 \\ -1, & \text{si } D > 100 \end{cases} \quad (1.9)$$

Cette définition de ω permet d'affecter au lien le maximum de pondération lorsqu'il se trouve à moins de 5m de la position GPS. Entre 5m et 100m, cette pondération diminue linéairement avec la distance. Au delà de 100m, la pondération est négative, car le lien est loin de la position GPS et la pondération permet ainsi de l'éliminer comme candidat.

Pondération de la position relative

Cette pondération se fait par rapport à un nœud du réseau de route. Elle permet de connaître la position du point GPS par rapport aux différents liens connectés à ce nœud. Sur la figure 1.17, P est la position GPS et B un nœud du réseau de routes. Les angles α_1 à α_4 sont les plus petits angles entre les différents liens connectés au nœud B et la ligne BP.

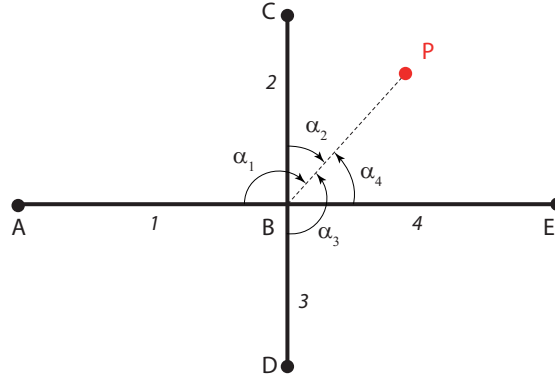


FIGURE 1.17 – Position relative par rapport aux liens reliés à un nœud

Ces angles sont suffisants pour connaître la position relative du point GPS P par rapport à ces différents liens. Dans le cas de la figure 1.17, les liens BC et BE sont les meilleurs candidats dans notre processus de sélection. En effet les angles pour ces deux liens α_2 et α_4 sont les plus petits parmi l'ensemble des angles associés aux liens connectés en B. On en déduit que le score de pondération doit être d'autant plus grand que l'angle α_i est petit. Ce score est défini alors

comme suit :

$$WS_{RP} = A_{RP} \cdot \cos(\alpha_i) \quad (1.10)$$

Où :

- WS_{RP} est le score de pondération de la position relative du véhicule par rapport au lien considéré,
- A_{RP} est le paramètre d'importance de WS_{RP} . Il est défini par l'équation (1.12).

Score de pondération total

Le score de pondération total TWS est la somme des trois scores de pondération WS_H , WS_{PD} et WS_{RP} définis précédemment par les équations (1.5), (1.8) et (1.10) respectivement :

$$TWS = A_H \cdot \cos(\Delta\beta') + A_{PD} \cdot \omega + A_{RP} \cdot \cos(\alpha_i) \quad (1.11)$$

Les valeurs $\cos(\Delta\beta')$, ω et $\cos(\alpha_i)$ varient toutes entre -1 et 1. Les paramètres d'importance A_H , A_{PD} et A_{RP} permettent de contrôler la contribution de chacun des trois critères à la pondération totale TWS . On calcule TWS pour chaque lien candidat. Le lien ayant le score le plus élevé est le lien correct sur lequel la position du véhicule est *matchée*.

Quddus propose dans [70] d'affecter les valeurs des paramètres d'importance de la manière suivante : On fixe une valeur pour la pondération de proximité A_{PD} puis on trouve les deux autres paramètres à l'aide des relations suivantes :

$$\begin{cases} A_H &= a \cdot A_{PD} \\ A_{RP} &= b \cdot A_{PD} \end{cases} \quad (1.12)$$

a et b sont des constantes positives qui traduisent notre connaissance *a priori* de la performance des capteurs et de la qualité de la carte numérique utilisée. A partir d'une expérience réalisée par Quddus avec 1800 points GPS obtenues sur un trajet dans la ville de Londres, plusieurs valeurs de a et b ont été testées. Pour chaque paire de valeurs, la précision du Map

matching obtenue a été évaluée. La valeur proposée pour A_{PD} était 10 qui est la moyenne de l'erreur horizontale du capteur GPS (10 mètres).

Il a constaté, après l'exécution de l'algorithme de Map matching, que l'erreur sur le matching était plus grande lorsque $a < b < 1$, autrement dit lorsque $A_H < A_{RP} < A_{PD}$. Il a conclu qu'il faut donner plus de poids au score de pondération du cap de véhicule WS_H par rapport au score de pondération de la position relative WS_{RP} , et que celui-ci, à son tour, doit avoir plus d'importance que le score de pondération de proximité A_{PD} . D'où la condition : $A_H > A_{RP} > A_{PD}$ ou encore $a > b > 1$.

D'après cette étude, les meilleurs résultats ont été trouvés avec $a = 3$ et $b = 2$ en utilisant le GPS/DR et avec $a = 2.5$ et $b = 2$ en utilisant le GPS tout seul. Dans les deux cas, la comparaison du cap du véhicule à la direction du lien a bien plus d'influence dans le choix du lien correct que l'information de la position relative ou de la distance du point GPS par rapport au lien.

Etapes de l'algorithme topologique amélioré

La figure 1.18 illustre la trajectoire d'un véhicule représentée par les points P^1 à P^5 relevés par un récepteur GPS . L'algorithme de Map matching topologique amélioré se déroule comme suit :

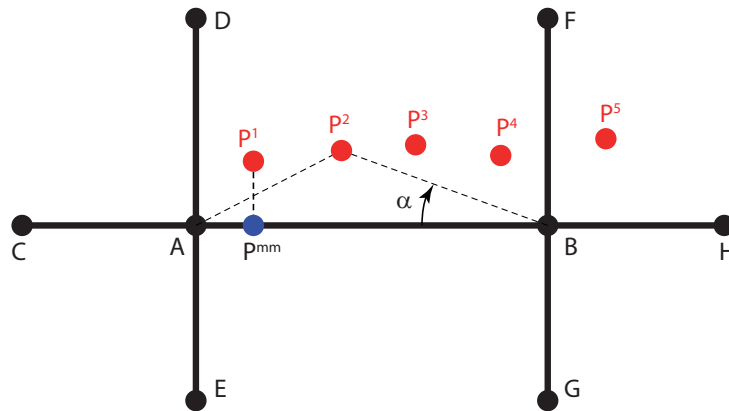


FIGURE 1.18 – Processus de l'algorithme topologique amélioré

1. Chercher le nœud le plus proche (sur la figure 1.18, c'est le nœud A) de la première position du véhicule P^1
2. Chercher tous les segments connectés au nœud A (ici AB, AC, AD, AE)

3. Calculer le score de pondération TWS pour chaque segment (formule 1.11), puis choisir le segment ayant le plus grand score (ici AB)
4. Déterminer la position par Map matching de P^1 en projetant ce point perpendiculairement sur AB , soit le point P^{mm}
5. Aller au point suivant (ici P^2). Calculer $\Delta\beta'$ et α correspondant au segment AB , tel que :
 - $\Delta\beta'$: écart entre la direction de AB et le cap du véhicule β mesuré en P^2 (voir figure 1.15 et formule 1.6)
 - α : angle de la position relative du point P^2 par rapport au nœud prochain B (voir figure 1.18)

On teste deux conditions :

- (a) Si $\Delta\beta' > 45^\circ$, on déduit que le véhicule a fait une manœuvre de rotation à droite ou à gauche
- (b) Si $\alpha > 90^\circ$, on déduit que le véhicule a dépassé la nœud B

Si l'une de ces conditions est vraie, on revient à l'étape 1 pour rechercher le nœud le plus proche. Sinon, le point P^2 est *matché* par projection perpendiculaire sur le segment AB (retour à l'étape 4)

L'idée de cet algorithme topologique est de continuer à se déplacer sur le même segment tant qu'on ne détecte pas de manœuvre de rotation. Cette manœuvre est indiquée soit par un écart angulaire $\Delta\beta' > 45^\circ$, soit un dépassement de la fin du segment lorsque $\alpha > 90^\circ$.

Bien que cet algorithme donne des résultats plus performants que l'algorithme de White [70], il existe plusieurs limitations qui dégradent la qualité de Map matching et qui seront discutées en détails dans le chapitre 3.

1.5 Analyse probabiliste

L'approche probabiliste en Map matching a été introduite par Honey et al. [32] dans le cas d'algorithmes reposant sur des capteurs proprioceptifs. Zhao dans [93] a étendu cette approche à la navigation par GPS. D'autres études ont aussi porté sur cette approche comme celle de Kim et al. [39], Pyo et al. [69] et Quddus [70].

Le principe est décrit par la figure 1.19. L'algorithme de Map matching lit les données de positionnement soit d'un capteur GPS seul ou associé à des capteurs proprioceptifs (GPS/DR). L'intégration des données mesurées peut être réalisée à l'aide d'un filtre de Kalman étendu

(EKF) comme proposé par Zhao et al. dans [92]. L'EKF fournit une estimation de la position (x, y) du véhicule sur la carte, sa vitesse v , son cap θ , ainsi que la variance des erreurs sur ces mesures : $\hat{\sigma}_x, \hat{\sigma}_y, \hat{\sigma}_v, \hat{\sigma}_\theta$ respectivement. A partir de ces données et celles de la carte numérique (liens, nœuds et variance de l'erreur sur la carte σ_{map}), l'algorithme de Map matching détermine le lien sur lequel le véhicule se trouverait ainsi que ses coordonnées sur ce lien : $\hat{x}_{lien}, \hat{y}_{lien}$ et l'incertitude associée à cette estimation $\hat{\sigma}_{mm}$.

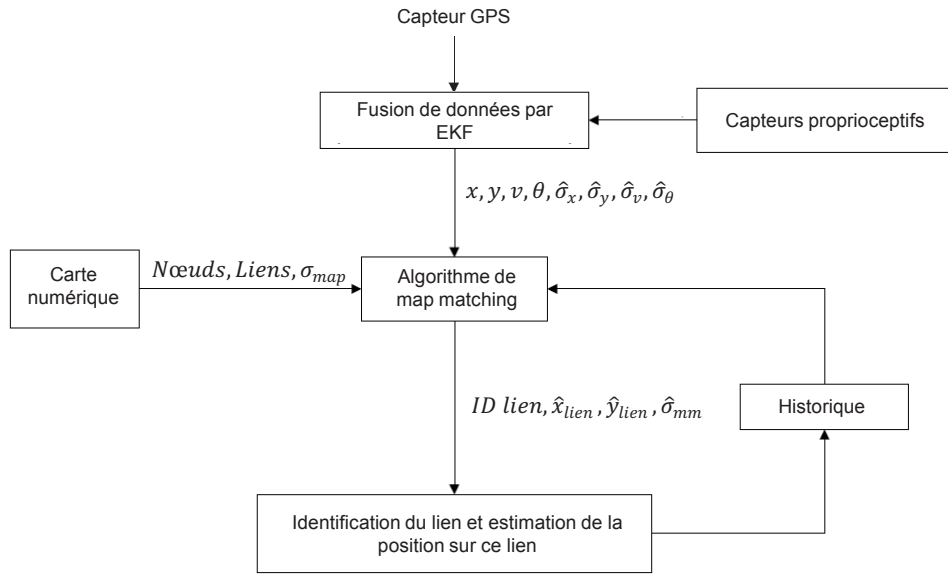


FIGURE 1.19 – Algorithme de Map matching selon l'approche probabiliste

Quddus propose dans [70] un algorithme de Map matching qui se déroule en deux étapes [70] :

- Map matching au niveau d'une jonction (MPJ) où l'algorithme détermine le lien où se trouve le véhicule,
- Map matching sur le lien trouvé par MPJ qu'on note SMPL (*Subsequent Map-matching Process on a Link*) qui se déplace sur le même lien jusqu'à détection d'une manœuvre de rotation, et à ce moment, on revient à MPJ pour identifier le nouveau lien.

Il est clair que l'étape MPJ de l'algorithme est cruciale car si le lien trouvé est incorrect, l'étape suivante SMPL donnera inévitablement des résultats erronés.

La méthode de détection mise en œuvre dans MPJ s'articule autour d'une approche probabiliste qui consiste à construire une zone de confiance elliptique autour de la position mesurée.

1.5.1 Détermination de la zone de confiance autour d'un point GPS

D'une manière générale, en supposant que l'écart-type d'une mesure est σ , la probabilité que la mesure soit dans une plage de 1σ est de 68,27%, si on élargit cette plage à 2σ , l'intervalle de confiance devient 95,45%. Avec 3σ , l'intervalle de confiance devient 99,73% et inclut presque la totalité des résultats .

Ce principe peut être étendu à 2 dimensions. Ainsi, pour une position GPS, la zone d'incertitude horizontale et verticale donne naissance à une ellipse [3] comme illustré sur la figure 1.20. L'orientation β de l'ellipse dépend de la corrélation entre x et y . S'ils ne sont pas corrélés, les axes de l'ellipse deviennent parallèles aux axes (x) et (y). De même, si la précision est la même sur les deux coordonnées, l'ellipse devient un cercle.

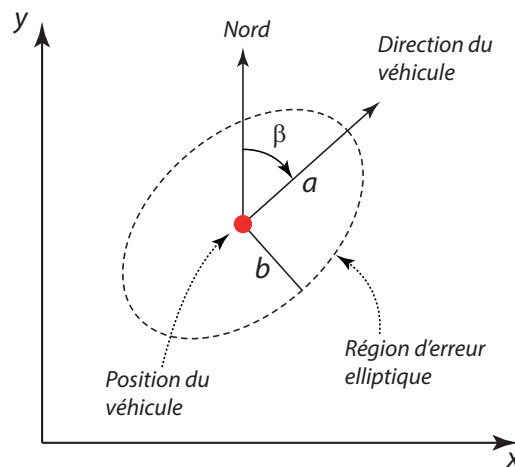


FIGURE 1.20 – Zone d'erreur autour de la position GPS

Considérons le cas général où la matrice de variance-covariance relative à la position est donnée par :

$$A = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} \\ \sigma_{yx} & \sigma_y^2 \end{bmatrix} \quad (1.13)$$

Où σ_x^2 et σ_y^2 sont les variances de l'erreur sur la position horizontale et verticale respectivement et σ_{xy} leur covariance.

Pour déterminer les axes de l'ellipse, on calcule le polynôme caractéristique de A donné par :

$$D(\lambda) = \det(A - \lambda I) = 0 \quad (1.14)$$

D'où :

$$(\sigma_x^2 - \lambda)(\sigma_y^2 - \lambda) - \sigma_{xy}^2 = 0 \quad (1.15)$$

Les valeurs propres de A sont les racines de cette équation caractéristique et qui sont :

$$\lambda = \frac{(\sigma_x^2 + \sigma_y^2) \pm \sqrt{(\sigma_x^2 + \sigma_y^2)^2 - 4(\sigma_x^2 \sigma_y^2 - \sigma_{xy}^2)}}{2} \quad (1.16)$$

La racine carrée de ces valeurs propres sont le grand axe a et le petit axe b de l'ellipse, soit :

$$a = \sqrt{\frac{(\sigma_x^2 + \sigma_y^2) + \sqrt{(\sigma_x^2 + \sigma_y^2)^2 - 4(\sigma_x^2 \sigma_y^2 - \sigma_{xy}^2)}}{2}} \quad (1.17)$$

$$b = \sqrt{\frac{(\sigma_x^2 + \sigma_y^2) - \sqrt{(\sigma_x^2 + \sigma_y^2)^2 - 4(\sigma_x^2 \sigma_y^2 - \sigma_{xy}^2)}}{2}} \quad (1.18)$$

L'angle β est déterminé par :

$$\beta = \frac{\pi}{2} - \frac{1}{2} \arctan\left(\frac{2\sigma_{xy}}{\sigma_x^2 - \sigma_y^2}\right) \quad (1.19)$$

Afin de compenser les diverses erreurs associées au GPS, telles que l'instabilité des orbites, la propagation atmosphérique, les multi-trajets et le bruit du récepteur, Zhao [93] propose que les valeurs de a et b soient multipliées par un facteur d'expansion $\hat{\sigma}_0$. Il propose aussi d'étendre cette région par un autre facteur multiplicatif qui traduit les imprécisions sur la carte numérique.

1.5.2 Déroulement de l'algorithme de Map matching probabiliste

La figure 1.21 illustre le déroulement de l'algorithme avec ses deux phases MPJ et SMPL. Comme expliqué précédemment, l'étape MPJ cherche à identifier le lien correct près d'une jonction. SMPL continue à *matcher* les positions suivantes sur ce lien jusqu'à détection d'une manœuvre de rotation.

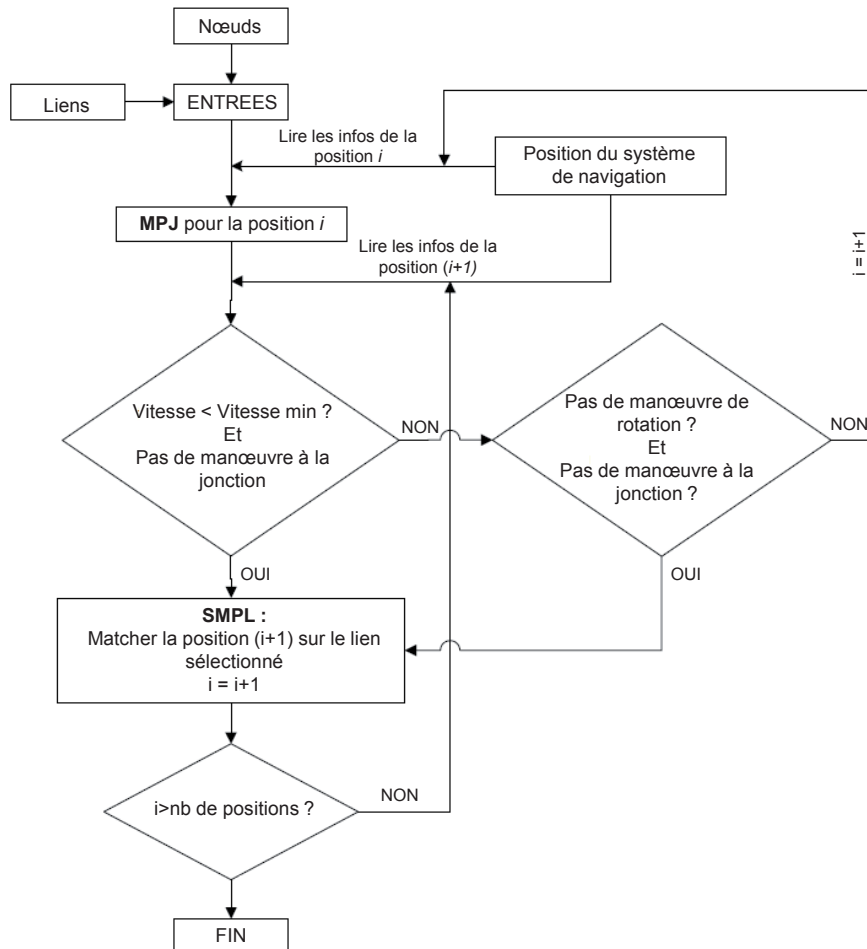


FIGURE 1.21 – Algorithme de Map matching probabiliste

Pour de faibles vitesses, le cap du véhicule fourni par le capteur GPS est fortement bruité. Qudus dans [70] propose une vitesse minimale pour que les mesures de cap soient acceptables de 10.8 km/h . Il a aussi proposé dans son étude qu'une manœuvre de rotation soit définie par les critères suivants :

- Une augmentation ou une diminution du cap du véhicule pendant une durée entre 2 et 5 secondes,

- Une différence en valeur absolue du cap entre la dernière position GPS et la position actuelle qui dépasse 30° ,
- Une différence en valeur absolue du cap entre l'avant dernière position GPS et la position actuelle qui dépasse 35° .

1.5.3 Détermination de la position du véhicule sur le lien

En supposant le lien correct identifié, la position du véhicule peut être déterminée de deux manières : soit utiliser la carte numérique avec la vitesse du véhicule issue des capteurs proprioceptifs, soit faire une projection perpendiculaire de la position GPS et GPS/DR sur le lien. Puisque l'erreur obtenue avec les deux méthodes est connue, une procédure est définie pour trouver la position finale du véhicule sur le lien.

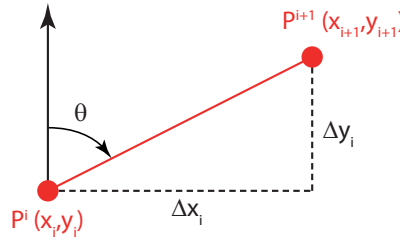


FIGURE 1.22 – Détermination de la position du véhicule avec la vitesse et le cap

Si on suppose P^i et P^{i+1} les positions du véhicule aux instants t et $t + 1$ respectivement (voir figure 1.22), et en supposant les coordonnées (x_i, y_i) du point P^i connues, on peut trouver les coordonnées de P^{i+1} en utilisant la vitesse v du véhicule et la direction du lien θ :

$$\begin{cases} x_{i+1} = x_i + \Delta x_i \\ y_{i+1} = y_i + \Delta y_i \end{cases} \quad (1.20)$$

avec :

$$\begin{cases} \Delta x_i = v \cdot \Delta t \cdot \sin \theta \\ \Delta y_i = v \cdot \Delta t \cdot \cos \theta \end{cases} \quad (1.21)$$

La seconde méthode s'appuie sur la position $P^S(x_S, y_S)$ fournie par le récepteur GPS (ou GPS/DR) figure 1.23. La projection perpendiculaire $P^{gps}(x_{gps}, y_{gps})$ sur le lien AB est donnée par :

$$\begin{cases} x_{gps} = \frac{(x_2 - x_1)[x_S(x_2 - x_1) + y_S(y_2 - y_1)] + (y_2 - y_1)(x_1 y_2 - x_2 y_1)}{(x_2 - x_1)^2 + (y_2 - y_1)^2} \\ y_{gps} = \frac{(y_2 - y_1)[x_S(x_2 - x_1) + y_S(y_2 - y_1)] + (x_2 - x_1)(x_1 y_2 - x_2 y_1)}{(x_2 - x_1)^2 + (y_2 - y_1)^2} \end{cases} \quad (1.22)$$

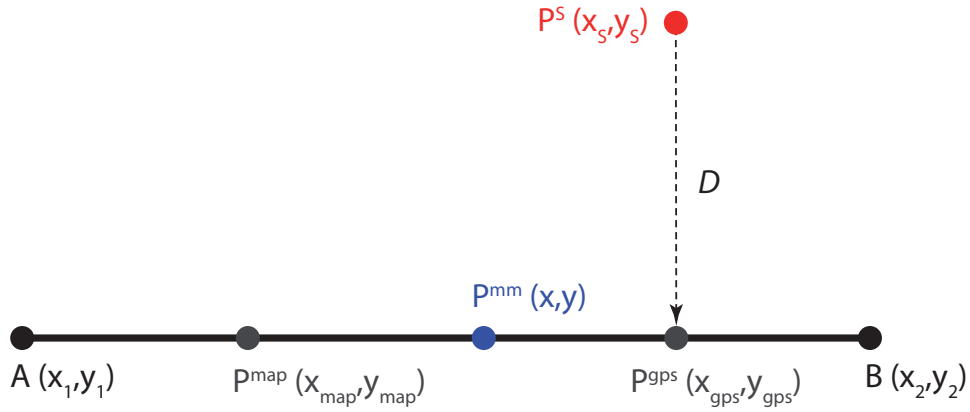


FIGURE 1.23 – Estimation de la position du véhicule sur le lien

Sur la figure 1.23, le point P^{map} représente la position calculée en utilisant la carte numérique et la vitesse (cf formule (1.20)). L'estimation de la position du véhicule P^{mm} est obtenue à partir des deux estimations P^{gps} et P^{map} .

Les composantes horizontales suivant l'axe des abscisses de P^{gps} et P^{map} peuvent être décrites par les équations suivantes :

$$x_{map} = x + \omega_{map,x} \quad (1.23)$$

$$x_{gps} = x + \omega_{gps,x} \quad (1.24)$$

Où x est la position horizontale, $\omega_{map,x}$ la composante horizontale de l'erreur associée à la position trouvée à partir de la carte digitale et de la vitesse, $\omega_{gps,x}$ l'erreur associée à la composante horizontale fournie par le récepteur GPS. On pose également $E(\omega_{map,x} \cdot \omega_{gps,x}) = \rho \cdot \sigma_{map,x} \cdot \sigma_{gps,x}$ avec ρ le coefficient de corrélation entre les deux erreurs et $\sigma_{map,x}$ et $\sigma_{gps,x}$

l'écart-type de $\omega_{map,x}$ et $\omega_{gps,x}$ respectivement.

L'estimation de x est la combinaison linéaire des deux mesures :

$$\hat{x} = k_1 x_{map} + k_2 x_{gps} \quad (1.25)$$

k_1 et k_2 sont deux coefficients indépendants de x . L'erreur d'estimation est dénotée par :

$$\tilde{x} = \hat{x} - x \quad (1.26)$$

Le critère utilisé pour définir les coefficients k_1 et k_2 est la minimisation du moment d'ordre 2 de \tilde{x} . De plus, l'estimation ne doit pas être biaisée, cela est vrai lorsque :

$$E(\tilde{x}) = E[k_1(x + \omega_{map,x}) + k_2(x + \omega_{gps,x})] = 0 \quad (1.27)$$

Étant donné que $E(\omega_{map,x}) = E(\omega_{gps,x}) = 0$ et que $E(x) = x$, l'équation (1.27) se réduit à :

$$k_1 = 1 - k_2 \quad (1.28)$$

Dans [70], la minimisation du moment d'ordre 2 de l'erreur d'estimation décrit par :

$$E(\tilde{x}^2) = E\left((k_1(x + \omega_{map,x}) + (1 - k_1)(x + \omega_{gps,x}) - x)^2\right) \quad (1.29)$$

se fait en cherchant la valeur de k_1 qui annule la dérivée partielle : $\frac{\partial E(\tilde{x}^2)}{\partial k_1}$. Cette valeur de k_1 est :

$$k_1 = \frac{\sigma_{gps,x}^2 - \rho\sigma_{map,x}\sigma_{gps,x}}{\sigma_{map,x}^2 + \sigma_{gps,x}^2 - 2\rho\sigma_{map,x}\sigma_{gps,x}} \quad (1.30)$$

On trouve k_2 à partir l'équation (1.28) :

$$k_2 = \frac{\sigma_{map,x}^2 - \rho\sigma_{map,x}\sigma_{gps,x}}{\sigma_{map,x}^2 + \sigma_{gps,x}^2 - 2\rho\sigma_{map,x}\sigma_{gps,x}} \quad (1.31)$$

L'estimation de la position horizontale est alors trouvée à partir des équations (1.25), (1.30) et (1.31) :

$$\hat{x} = \frac{\sigma_{gps,x}^2 - \rho\sigma_{map,x}\sigma_{gps,x}}{\sigma_{map,x}^2 + \sigma_{gps,x}^2 - 2\rho\sigma_{map,x}\sigma_{gps,x}}.x_{map} + \frac{\sigma_{map,x}^2 - \rho\sigma_{map,x}\sigma_{gps,x}}{\sigma_{map,x}^2 + \sigma_{gps,x}^2 - 2\rho\sigma_{map,x}\sigma_{gps,x}}.x_{gps} \quad (1.32)$$

La variance de cette estimation est trouvée après développement de l'équation (1.29) [70] :

$$\sigma_x^2 = E(\hat{x}^2) = \frac{\sigma_{map,x}^2\sigma_{gps,x}^2(1-\rho^2)}{\sigma_{map,x}^2 + \sigma_{gps,x}^2 - 2\rho\sigma_{map,x}\sigma_{gps,x}} \quad (1.33)$$

De la même manière on calcule par les formules suivantes, l'estimation de la position horizontale suivant l'axe des ordonnées et sa variance :

$$\hat{y} = \frac{\sigma_{gps,y}^2 - \rho\sigma_{map,y}\sigma_{gps,y}}{\sigma_{map,y}^2 + \sigma_{gps,y}^2 - 2\rho\sigma_{map,y}\sigma_{gps,y}}.y_{map} + \frac{\sigma_{map,y}^2 - \rho\sigma_{map,y}\sigma_{gps,y}}{\sigma_{map,y}^2 + \sigma_{gps,y}^2 - 2\rho\sigma_{map,y}\sigma_{gps,y}}.y_{gps} \quad (1.34)$$

$$\sigma_y^2 = \frac{\sigma_{map,y}^2\sigma_{gps,y}^2(1-\rho^2)}{\sigma_{map,y}^2 + \sigma_{gps,y}^2 - 2\rho\sigma_{map,y}\sigma_{gps,y}} \quad (1.35)$$

l'estimateur de la position horizontale ainsi obtenue est optimal au sens de la variance minimale.

1.5.4 Calcul de la variance des erreurs

Les variances des erreurs de la mesure GPS $\sigma_{gps,x}^2$ et $\sigma_{gps,y}^2$ sont obtenues à partir de la matrice de variance-covariance associée au système de navigation GPS tel que décrit par Zhao dans [93]. Les variances $\sigma_{map,x}^2$ et $\sigma_{map,y}^2$ sont, en revanche, calculées à partir des équations (1.21) illustrées par la figure 1.22 qui décrit les variations de positions $(\Delta x_i, \Delta y_i)$ associées à la vitesse v et à la direction θ . En supposant le pas temporel $\Delta t = 1$, l'incertitude associée aux variations de positions est donnée par :

$$\begin{cases} d(\Delta x_i) = \sin \theta . dv + v \cos \theta . d\theta \\ d(\Delta y_i) = \cos \theta . dv - v \sin \theta . d\theta \end{cases} \quad (1.36)$$

On suppose ici que cette incertitude est distribuée suivant une distribution gaussienne, dont les écarts-types des composantes des erreurs horizontales et verticales $\omega_{map,x}$ et $\omega_{map,y}$, sont donnés par :

$$\begin{cases} \sigma_{map,x} = \sqrt{(\sin \theta . \Delta v)^2 + (v \cos \theta . \Delta \theta)^2} \\ \sigma_{map,y} = \sqrt{(\cos \theta . \Delta v)^2 + (-v \sin \theta . \Delta \theta)^2} \end{cases} \quad (1.37)$$

Où Δv est l'incertitude associée à la vitesse du véhicule obtenue de la matrice de variance-covariance de position, et $\Delta \theta$ est l'incertitude associée à la direction obtenue du lien des données de la carte numérique.

Notons que dans le cas particulier où il n'existe pas de corrélation entre la mesure GPS et la mesure qui utilise la carte numérique et la vitesse, on a $E(\omega_{map,x} . \omega_{gps,x}) = 0$ et $\rho = 0$, et dans ce cas, l'estimation optimale de la position décrite par les équations (1.32) et (1.34) s'écrit :

$$\hat{x} = \frac{\sigma_{gps,x}^2}{\sigma_{map,x}^2 + \sigma_{gps,x}^2} . x_{map} + \frac{\sigma_{map,x}^2}{\sigma_{map,x}^2 + \sigma_{gps,x}^2} . x_{gps} \quad (1.38)$$

$$\hat{y} = \frac{\sigma_{gps,y}^2}{\sigma_{map,y}^2 + \sigma_{gps,y}^2} . y_{map} + \frac{\sigma_{map,y}^2}{\sigma_{map,y}^2 + \sigma_{gps,y}^2} . y_{gps} \quad (1.39)$$

La variance de ces estimations devient (cf équations (1.33) et (1.35)) :

$$\sigma_x^2 = \frac{\sigma_{map,x}^2 \sigma_{gps,x}^2}{\sigma_{map,x}^2 + \sigma_{gps,x}^2} \quad (1.40)$$

$$\sigma_y^2 = \frac{\sigma_{map,y}^2 \sigma_{gps,y}^2}{\sigma_{map,y}^2 + \sigma_{gps,y}^2} \quad (1.41)$$

Ou encore :

$$\frac{1}{\sigma_x^2} = \frac{1}{\sigma_{map,x}^2} + \frac{1}{\sigma_{gps,x}^2} \quad (1.42)$$

$$\frac{1}{\sigma_y^2} = \frac{1}{\sigma_{map,y}^2} + \frac{1}{\sigma_{gps,y}^2} \quad (1.43)$$

On voit clairement que la variance de l'estimation est toujours inférieure à la variance des deux mesures. Ce qui veut dire que la combinaison de ces deux mesures améliore la précision de la position estimée. En outre, en observant les équations (1.38) et (1.39), on remarque bien que la position estimée est une somme pondérée des deux mesures, et que les coefficients de pondération favorisent la mesure ayant l'incertitude la plus faible.

1.5.5 Algorithme détaillé de Map matching reposant sur l'approche probabiliste

Comme indiqué précédemment, l'algorithme de Map matching reposant sur l'approche probabiliste se déroule en deux étapes : MPJ et SMPL. Dans ce qui suit, nous décrivons les différentes actions sous-jacentes à chaque étape.

Etape MPJ

L'étape MPJ dépend de l'espace de recherche, de la proximité de la position mesurée au lien, de l'orientation relative du véhicule par rapport au lien, de la topologie de la carte (connexion entre les différents liens) et de l'historique de navigation du véhicule.

Si l'on considère la position GPS représentée par la figure 1.24, l'algorithme MPJ se déroulera comme suit :

- Définir l'espace de recherche en fonction de la qualité de la mesure de position (indiqué par l'ellipse d'erreur),
- Identifier tous les segments de routes appartenant à l'espace de recherche (ici 2,3,4,5,6,7 et 8),
- Filtrer les liens candidats en comparant le cap du véhicule et la direction de chaque lien. Si cette différence est inférieure à 45°, le lien est retenu. Dans notre cas, on retient les

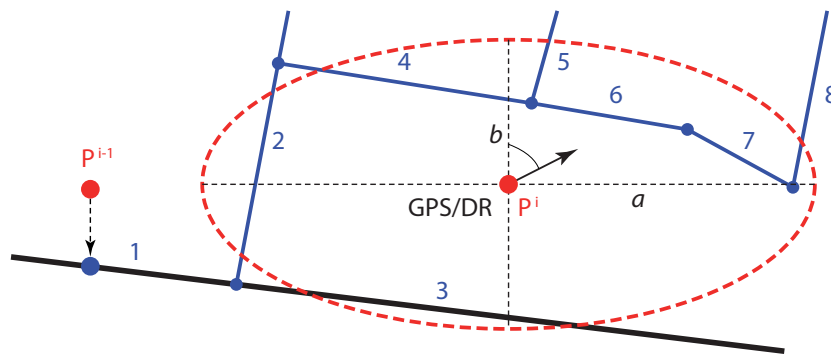


FIGURE 1.24 – Matching sur une jonction (MPJ)

liens : 2,3,4,5,6 et 7,

- Utiliser l'historique de navigation du véhicule et la topologie du réseau pour éliminer davantage de liens. Dans notre cas, étant donné que la position précédente du véhicule a été matchée au lien 1 et vu la topologie du réseau de routes, la position actuelle ne pourrait être fixée que sur les liens 2 ou 3. Si la position précédente n'est pas connue, l'algorithme passe à l'étape suivante,
- Choisir le lien final en se référant à la proximité et à l'orientation relative de la position mesurée par rapport au lien. Le lien qui respecte ces conditions est le lien 3,
- La position du véhicule sur le lien est calculée en utilisant les équations (1.38) et (1.39).

Etape SMPL

L'étape SMPL dépend de la différence entre la direction du lien et le cap du véhicule, la distance jusqu'à la jonction suivante D , la vitesse et la vitesse de rotation du gyromètre (si disponibles). Durant cette étape, les positions mesurées sont *matchées* sur le lien identifié par MPJ si les conditions suivantes sont remplies :

- Absence de détection de manœuvre de rotation,
- La distance jusqu'à la jonction suivante D est supérieure à 10m,
- La vitesse de rotation du gyromètre est inférieure à $15^\circ/s$.

La position du véhicule sur le lien est calculée en utilisant les équations (1.38) et (1.39).

1.6 Autres approches d'analyse

1.6.1 Analyse par logique floue

La logique floue est une technique qui est largement employée pour traiter les termes qualitatifs, les propositions linguistiques vagues et l'intervention humaine [93]. En logique floue, les termes vagues comme grand, petit, moyen, ... peuvent être formulés mathématiquement en utilisant des ensembles flous. Des règles représentant l'expertise et le savoir de l'expert sont utilisées pour tirer des inférences à travers un processus de raisonnement approximatif.

En Map matching, l'identification du lien correct sur lequel le véhicule se trouve est un processus qualitatif de prise de décision qui implique un degré d'ambiguïté. Les entrées d'un algorithme de Map matching utilisant la logique floue peuvent être la position GPS ou GPS/DR. Le principe de cette approche est de construire plusieurs règles de connaissance sous forme de SI-ALORS qui prennent en considération plusieurs entrées comme la vitesse du véhicule, son cap, l'historique de sa trajectoire et la connectivité et orientation des liens du réseau de routes.

Durant les dernières années, les algorithmes de Map matching reposant sur la logique floue étaient l'objet de plusieurs publications et travaux de recherche, on cite à titre d'exemple Zhao [93], Kim et al [38], Syed et Cannon [82] et Quddus [71].

La théorie de la logique floue a été introduite par Zadeh en 1965 [90] avec l'intention de traiter les problèmes faisant appel aux connaissances exprimées en termes linguistiques vagues. Pour représenter des nuances telles que : la vitesse du véhicule est faible, la différence entre le cap et la direction de la route est grande, ... des concepts comme le degré d'appartenance et le degré de vérité ont été introduits. Une étude exhaustive de la théorie de la logique floue peut être trouvée dans [90, 91, 61, 81]. Les grandes lignes sont illustrées par le système d'inférence de Mamdani de la figure 1.25.

Considérons la règle de connaissance floue suivante : « Si la distance entre la position GPS et le lien est grande alors la possibilité de matcher la position GPS au lien est faible ». La variable d'entrée pour cette règle concerne "*la distance entre le position GPS et le lien*", et son sous-ensemble flou est "*grande*". La variable de sortie donne "*la possibilité de matcher le point GPS au lien*", et son sous-ensemble flou est "*faible*". Puisque les sous-ensembles flous décrivent des concepts vagues, la vérité de toute proposition, comme la distance entre le position GPS et le lien, devient une question de degré d'appartenance. Ceci est assuré par la *fuzzification* de la variable d'entrée en utilisant une *fonction d'appartenance*. Une telle fonction indique comment

un point de l'espace d'entrée (ici la distance entre la position GPS et le lien) est associé à un degré d'appartenance à un ensemble flou entre 0 et 1. Plusieurs formes de ces fonctions existent dans la littérature : triangulaire, trapézoïdal, gaussienne, ... Le choix de la forme se fait souvent d'une façon empirique.

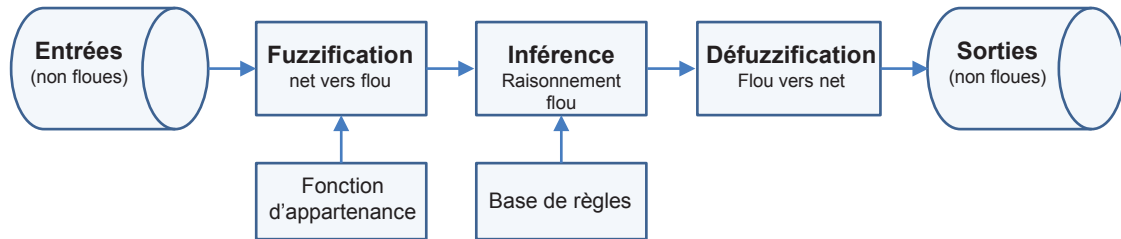


FIGURE 1.25 – Système d'inférence flou de Mamdani

Une fois la *fuzzification* faite pour toutes les entrées, on formule des règles de connaissance floues sous forme de SI-ALORS. La condition qui suit SI s'appelle *l'antécédent* ou *la prémisse*. La partie qui suit ALORS s'appelle *la conséquence* ou *la conclusion*. La sortie de chaque règle est aussi un ensemble flou qui est déterminé avec une méthode d'implication : 1) *min* (minimum) ou *prod* (produit) si l'antécédent est formé par un ET comme par exemple : « Si la distance est faible ET la différence de direction est grande », 2) *max* ou *probor* (*probabilistic OR* : OU probabiliste) si l'antécédent est formé par un OU [61].

Les ensembles flous résultant de chaque règle sont ensuite combinés en un seul ensemble flou en utilisant une des méthodes d'agrégation : 1) *max* (maximum), 2) *probor* (OU probabiliste), 3) *somme* [61].

La dernière étape est le processus de *défuzzification*. L'entrée de ce processus est le résultat de la méthode d'agrégation, la sortie est une valeur numérique nette qui représente la décision. Plusieurs méthodes sont utilisables pour la défuzzification comme : l'appartenance maximale, la méthode du centroïde, la méthode des moyennes pondérées, ...

Le figure 1.25 illustre un système d'inférence flou de type Mamdani [61]. Sugeno a défini dans [81] un autre système d'inférence qui diffère de celui de Mamdani en proposant le remplacement de la défuzzification par l'utilisation de la moyenne pondérée pour le calcul de la valeur nette de la sortie.

1.6.2 Analyse par la théorie des croyances

Dans [65], les auteurs proposent un algorithme de Map matching qui utilise la théorie des croyances pour gérer les hypothèses multiples. Cette théorie, connue aussi sous le nom de la théorie de Dempster-Shafer ou la théorie de l'évidence est un outil qui permet de traiter les incertitudes, les imprécisions ou le manque d'information.

L'approche proposée consiste à localiser l'intervalle de connaissance à partir des mesures, puis sélectionner un ensemble de segments candidats dans une région rectangulaire. Le meilleur segment candidat est choisi par application d'une règle de décision de la fonction de croyance.

Une transformation est appliquée aux routes qui sont alors représentées sous forme rectangulaire. L'analyse par intervalle est appliquée au modèle et à la mesure. Ainsi, un intervalle est défini pour chaque mesure, les dimensions dépendant de la variance de la mesure considérée. Ceci résulte en un pavé défini pour l'état noté $[x_k]$ et un pavé défini pour la position mesurée noté $[z_k]$.

Un ensemble de segments candidats R_k est construit par les routes rectangulaires $[r]$ qui respectent la règle $[x_k] \cap [r] \neq \emptyset$.

Le principe du Map matching exposé consiste à calculer à chaque instant k , une fonction de masse notée m^{R_k} en utilisant la topologie et la similarité comme critère. Cette fonction représente le degré de croyance que le véhicule soit présent sur chacune des routes incluses dans R_k .

L'algorithme se déroule en cinq étapes :

1. *Initialisation*
2. *Prédiction* : Les pavés correspondants à chaque route de R_k notés $\{[x_k^i]\}_{i=1}^{n_{R_k}}$ sont prédits par le modèle d'état donnant naissance aux pavés $\{[x_{k+1}^i]\}_{i=1}^{n_{R_k}}$
3. *Correction GPS* : Le pavé de la mesure GPS $[z_{k+1}]$ est utilisé pour corriger la prédiction en faisant une intersection des deux pavés : celui de la prédiction et celui de la mesure.
4. *Correction SIG (Système d'Information Géographique)* : si le centre du pavé $[x_k^i]$ dépasse la fin d'un segment, toutes les routes S^i reliées à la route r^i sur laquelle existe le pavé sont ajoutés à l'ensemble des routes candidates $R_{k+1} = R_k \cup S^i$, sinon cet ensemble reste invariable : $R_{k+1} = R_k$

La fonction de masse $m^{R_{k+1}}$ est ensuite calculée pour le nouvel ensemble des segments candidats R_{k+1} . Le segment sélectionné est celui ayant la plus grande valeur de la fonction de plausibilité tirée de $m^{R_{k+1}}$.

L'étude expérimentale menée dans [65] sur une intersection de routes ainsi que sur un trajet réel montre que cette méthode arrive à améliorer la localisation à partir de mesures issues d'un capteur GPS, d'un gyromètre et d'un odomètre. Cependant la robustesse de cette méthode n'a pas été évaluée.

1.6.3 Analyse par l'utilisation d'une carte précise

Les auteurs dans [84] proposent une méthode de la localisation à l'aide d'une nouvelle description du réseau de routes, nommée "Carte améliorée" ou Emap (*Enhanced map*). Une description détaillée de cette carte peut être trouvée dans [17]. Une carte Emap se différencie d'une carte classique par le fait qu'elle intègre une description géométrique, sous forme d'une série de clothoïdes, de la ligne au milieu de chaque voie d'une route. Une route, dans ce cas, est un ensemble de voies ayant un sens de déplacement défini. Il n'existe pas encore de modèle pour les rond points avec cette approche.

Les clothoïdes sont caractérisées par les coordonnées, la direction et la courbure à leur point d'origine, ainsi que par le taux de variation de leur courbure le long d'une abscisse curviligne. Le passage aux coordonnées cartésiennes x, y fait appel à des intégrales impropres connues sous le nom d'Intégrales de Fresnel calculées numériquement par morceaux.

Dans ce sens, les cartes Emap sont plus proches de la réalité que les cartes classiques qui modélisent les routes courbées par une suite contiguë de segments linéaires. Cependant, leur construction nécessite des mesures sur le terrain et un traitement postérieur pour extraire les clothoïdes (voir [17]).

Le Map matching proposé dans [84] repose sur la fusion des données GPS, EGNOS et proprioceptifs (un gyromètre et un odomètre) avec les données géométriques issues d'une carte Emap. Ce processus est réalisé par un filtre particulière.

Dans l'étape d'initialisation, les particules sont créées selon une distribution gaussienne avec la première position GNSS valide comme moyenne. Le choix initial de la voie utilise la proximité comme seul critère sauf si d'autres informations *a priori* sont disponibles. Dans l'étape

de prédiction, les données du gyromètre et de l'odomètre sont utilisées pour propager le nuage des particules sur la voie sélectionnée. Les équations de propagation prennent aussi en compte les informations géométriques de la voie provenant de l'Emap. Dans l'étape de correction, on vérifie si chaque particule se trouve toujours sur la même voie en comparant sa position actuelle par rapport à la largeur et la longueur de la voie. Si elle se trouve en dehors, on cherche dans la base de données de l'Emap une nouvelle voie en utilisant la proximité comme critère. Si aucune route n'est trouvée, la particule est éliminée. Après ré-échantillonnage des particules, une correction de la position est faite par les mesures GNSS si elles sont disponibles.

L'évaluation de l'algorithme proposé, sur des données réelles, a montré sa contribution à définir la voie du véhicule avec précision. Cependant, la difficulté de la construction de la carte Emap reste le principal inconvénient qui limite l'application de cette méthode à des zones d'intérêt spécifiques.

1.6.4 Analyse locale et globale

Les algorithmes de Map matching peuvent aussi être classés en deux catégories : *algorithmes locaux/incrémentaux* et *algorithmes globaux*.

Un algorithme local/incrémental suit une stratégie incrémentale qui consiste à étendre séquentiellement la solution trouvée sur la portion déjà *matchée* de la trajectoire. Ce type d'algorithme essaie de trouver localement sur la carte un nœud ou un lien optimal en se référant à la distance, l'orientation ou la similarité. Les approches géométriques *point-to-point*, *point-to-curve* ainsi que l'approche topologique et statistique entrent toutes dans ce cadre.

En général, un algorithme local/incrémental est rapide et s'adapte à des fréquences d'échantillonnage élevées (par exemple 1 à 5 secondes entre deux mesures de position). De ce fait, ce type d'algorithme est orienté aux applications où l'analyse en temps réel est requise. Cependant, la performance de ces algorithmes diminue considérablement lorsque la fréquence des mesures diminue. Dans ce cas, le phénomène du « saut d'arc » (*arc-skipping*) devient marquant [29]. En effet, lorsque deux points GPS sont séparés par une longue distance, il devient difficile ou parfois impossible de connaître avec précision la séquence précise des segments de routes empruntés par le véhicule entre ces deux points car plusieurs possibilités se présentent [59]. En outre, ces algorithmes cherchent à matcher uniquement une petite portion de la trajectoire et ne considèrent souvent dans ce processus que la dernière position *matchée*. Chose qui augmente la

probabilité d'erreur si la dernière position n'est pas précise.

Le second groupe d'algorithmes, dits globaux, cherche à aligner la totalité de la trajectoire sur le réseau de routes en considérant ce dernier comme un graphe. Cette approche peut augmenter la précision du matching, mais au détriment d'un coût de calcul plus élevé. Les algorithmes globaux existant reposent toujours sur une *mesure de distance/similarité* comme la *distance de Fréchet moyenne* par exemple. Ils ont aussi leurs défauts comme leur dépendance de l'aspect spatial de la trajectoire sans prise en compte de ses contraintes temporelles.

Brakatsoulas compare dans [23] un algorithme incrémental et deux algorithmes globaux. Son algorithme incrémental utilise un système de pondération reposant sur la distance entre position GPS et lien candidat et sur la similarité de direction entre les deux. L'amélioration qu'il propose par rapport à l'existant est le processus d'anticipation locale (*local look-ahead*) qui consiste à calculer le poids de chaque route possible partant d'un point GPS et passant par un ou plusieurs points GPS suivants. Cet algorithme est notamment efficace pour les basses fréquences de mesure où plusieurs alternatives d'itinéraire se présentent entre deux points GPS consécutifs.

Le premier algorithme global proposé par Brakatsoulas cherche à trouver sur le réseau routier la courbe qui s'approche le plus à la trajectoire du véhicule en utilisant la *distance de Fréchet* et la *distance de Fréchet faible*. Il repose sur l'algorithme proposé par Alt et al. [1] qui réalise un Map matching global avec un coût de calcul $O(mn \log^2 mn)$ où m est le nombre des sommets et des liens dans le réseau routier, et n le nombre des positions du véhicule. En utilisant la *distance de Fréchet faible*, Brakatsoulas est arrivé à diminuer le coût de calcul à $O(mn \log mn)$.

1.7 Conclusion

Nous avons présenté dans ce chapitre une analyse des différents algorithmes de Map matching existant dans la littérature. Plusieurs méthodes ont été abordées allant des techniques simples comme les algorithmes géométriques point-à-point et point-à-courbe, jusqu'aux approches complexes qui font appel à la théorie de probabilité et la logique floue. Une brève description des ces algorithmes et leur performance a été aussi présentée.

Nous avons identifié les contraintes et limitations de ces algorithmes que nous pouvons résumer dans les points suivants :

- *Les problèmes associés au Map matching de la première position du véhicule* : Une importance particulière doit être accordée au premier matching car un matching initial imprécis peut causer un mauvais choix du lien initial, d'où des erreurs de matching sur les positions postérieures. Plusieurs des algorithmes traités, proposent un sous-algorithme particulier pour identifier la première position et un autre pour les positions suivantes.
- *Les problèmes de Map matching dans des environnements urbains denses* : Dans de tels réseaux de routes, les liens sont condensés et très rapprochés. Il devient difficile de faire un matching correct dans les jonctions sous forme Y par exemple, les ronds-points et les routes parallèles très proches. En plus, dans ces environnements, la précision des capteurs GPS diminue à cause des phénomènes des multi-trajets, du masquage fréquents des satellites et de la dilution horizontale de précision (HDOP).
- *La fréquence des mesures GPS* : Tous les algorithmes abordés fonctionnent à des fréquences de mesures GPS élevées (entre 1 à 10s de délai entre mesures). Ces algorithmes deviennent quasiment inutilisables si une longue période sépare les mesures (1 à 3 minutes par exemple), c'est notamment le cas du tracking à distance des véhicules par GPRS ou dans les environnement urbains où les satellites GPS peuvent être masqués pendant plusieurs minutes. L'éloignement des points mesurés rend difficile la connaissance de la trajectoire réelle empruntée par le véhicule, surtout si ce dernier fait plusieurs détours consécutifs. Le recours aux capteurs proprioceptifs ne résout pas le problème car leur précision se dégrade rapidement dans le temps.

Un autre aspect important des algorithmes de Map matching est le temps de calcul. Les algorithmes dits incrémentaux comme les algorithmes géométriques et topologiques consomment moins de temps de calcul, car la détermination du lien actuel dépend de la position GPS mesurée et d'une ou deux positions antérieures. En revanche, Les algorithmes globaux traitent la totalité de la trajectoire et essayent de trouver sur le réseau de routes le chemin le plus proche qui minimise une distance de référence comme la distance de Fréchet. Ces algorithmes consomment plus de temps de calcul et sont souvent adaptés au Map matching *off-line*.

Chapitre 2

Filtrage multi-modèles

2.1 Introduction

Dans le premier chapitre, on a décrit les approches de map matching existantes. Ces méthodes reposent principalement sur la position fournie par les capteurs GPS ou proprioceptifs. Il est donc crucial d'obtenir la meilleure estimation de cette position pour assurer un map matching précis. Dans ce chapitre, on s'intéressera donc au problème de l'estimation de la position du véhicule sachant que celui-ci peut effectuer des manœuvres le long de son trajet telles que l'accélération, la décélération, des virages à gauche ou à droite. Dans ce cas, l'utilisation d'un filtre de Kalman avec un modèle unique n'est pas la meilleure solution. En effet, le modèle du mouvement du véhicule change au cours du temps d'une manière imprévisible et on est alors dans le cadre d'un problème de poursuite (*target tracking*). Dans ce type d'application, en général, les entrées de commande associées aux manœuvres de l'objet observé ne sont pas connues. De plus, les paramètres du système, qui sont supposés connus, peuvent aussi changer en fonction de la nature de la manœuvre.

Plusieurs techniques sont employées pour estimer l'état de ces systèmes, on cite par exemple : l'estimation robuste qui cherche à établir un filtre qui fournit des performances assez bonnes pour une large plage de paramètres du modèle (Filtre de Kalman robuste ou H_2 , filtre H_∞), ou l'estimation d'état adaptative qui cherche à estimer simultanément l'état inconnu x_k et les paramètres inconnus du système, notamment les paramètres du bruit. Plusieurs approches entrent dans ce cadre telles que : le filtrage joint de l'état et des paramètres avec un filtre EKF, le réglage en ligne du bruit et enfin l'estimation multi-modèles (MM).

L'approche MM est une méthode, robuste et adaptative utilisée pour résoudre plusieurs pro-

blèmes pratiques. Elle a plusieurs applications comme la poursuite de cibles, la détection et l'identification de défauts. Elle est applicable dans des domaines variés comme par exemple le traitement du signal en biomédical et les systèmes de navigation.

L'approche MM est une méthode de résolution d'un problème d'estimation d'une variable aléatoire (l'état) à partir d'une observation aléatoire (la mesure). Elle s'appuie sur un ensemble de filtres élémentaires opérant en parallèle à chaque instant. Chaque filtre repose sur un modèle différent. L'estimation globale est obtenue par une combinaison des estimations de chaque filtre.

La méthode multi-modèles s'est développée en trois générations : dans la première appelée *approche MM autonomes*, les filtres élémentaires fonctionnent de façon indépendante. Ces filtres peuvent interagir les uns avec les autres en partageant des informations ; cela a donné naissance à la deuxième génération connue sous le nom de *MM interactifs*. La troisième génération dite *approche MM à structure variable* ou *VSMM* hérite des deux premières générations et met en œuvre en plus un ensemble de modèles variants qui s'adaptent à l'environnement. Pour les deux premières générations, l'ensemble des modèles est invariant d'où l'appellation *MM à structure fixe* ou *FSMM*. La seconde et la troisième génération sont particulièrement adaptées à la poursuite de cibles [8] et ceci non seulement pour décrire les différents types de mouvements (par exemple : missiles, véhicules terrestres, navires, ...[50]) mais aussi les systèmes de mesure [51] ainsi que les statistiques sur le bruit de mouvement et les erreurs de mesures [49].

Un autre domaine d'application important des méthodes multi-modèles est la détection et l'identification de défauts, où les différents capteurs/actionneurs ou défauts du système sont représentés par différents modèles comme dans [86, 12]. D'autres applications des méthodes multi-modèles sont décrites dans [45] ainsi que dans les différentes références qui y sont citées comme : la linéarisation par morceaux de systèmes non linéaires, les systèmes variables dans le temps, le contrôle adaptatif, le guidage et la navigation, le filtrage adaptatif et l'identification, la commande de vol, le traitement des signaux sismiques, le traitement des signaux biomédicaux et la géodésie.

Dans ce chapitre nous allons décrire le principe de l'estimation d'une manière générale, puis nous passerons en revue les différentes approches de l'estimation multi-modèles tout en soulignant les points forts et faibles de chacune.

2.2 Estimation et filtrage d'état

Un problème d'estimation d'un paramètre x peut être décrit de la manière suivante :

Etant données les mesures :

$$z_j = h(j, x, w_j) \quad j = 1, \dots, k \quad (2.1)$$

faites en présence d'un bruit w_j , on désire trouver une fonction de k observations telle que :

$$\hat{x}_k \triangleq \hat{x}(k, Z_k) \quad (2.2)$$

où Z_k représente les observations :

$$Z_k \triangleq \{z_j\}_{j=1}^k \quad (2.3)$$

de la valeur de x .

La fonction \hat{x} est appelée *Estimateur*. La valeur de cette fonction \hat{x}_k est l'*estimée*. L'erreur d'estimation correspondante à l'estimée \hat{x}_k est :

$$\tilde{x}_k = x_k - \hat{x}_k \quad (2.4)$$

L'estimation paramétrique est considérée comme *un problème d'estimation statique*, tandis que l'estimation d'état est considérée comme *un problème d'estimation dynamique*.

Il existe deux modèles que l'on utilise dans l'estimation d'un paramètre :

- *Non aléatoire* : Il existe une valeur inconnue réelle x_0 . Ce modèle est aussi appelé *non bayésien* ou *approche Fisher*.
- *Aléatoire* : Le paramètre est une variable aléatoire avec une densité de probabilité *a priori* $p(x)$. Une réalisation de x selon $p(x)$ est supposée constante pendant le processus de mesure. Cela est aussi appelé *approche bayésienne*.

Selon l'approche bayésienne, on part de la densité de probabilité *a priori* du paramètre pour obtenir la densité de probabilité *a posteriori* en utilisant la célèbre formule de Bayes :

$$p(x|Z) = \frac{p(Z|x)p(x)}{p(Z)} = \frac{1}{c} p(Z|x)p(x) \quad (2.5)$$

Où c est une constante de normalisation ne dépendant pas de x .

L'approche non bayésienne, en revanche, n'a pas de densité de probabilité *a priori* ou *a posteriori*. On dispose uniquement de la densité de probabilité des mesures conditionnées par le paramètre, c'est ce qu'on appelle *la fonction de vraisemblance* du paramètre :

$$\Lambda_k \triangleq p(Z_k|x) \quad (2.6)$$

qui mesure le degré de "vraisemblance" du paramètre par rapport aux mesures obtenues. Cette fonction de vraisemblance joue un rôle similaire à la densité de probabilité *a posteriori* dans l'approche bayésienne.

2.2.1 Estimateurs du maximum de vraisemblance et du maximum *a posteriori*

Estimateur du maximum de vraisemblance

Une méthode courante d'estimation d'un paramètre non aléatoire est la méthode du maximum de vraisemblance qui repose sur la maximisation de la fonction de vraisemblance définie par (2.6). Cet estimateur est appelé *Estimateur à maximum de vraisemblance (Maximum likelihood (ML) estimator)*

$$\hat{x}^{\text{ML}}(Z) = \arg \max_x \Lambda_Z(x) = \arg \max_x p(Z|x) \quad (2.7)$$

$\hat{x}^{\text{ML}}(Z)$ est une fonction de l'ensemble des observations aléatoires Z . C'est donc une variable aléatoire. L'estimateur ML est la solution de l'équation de vraisemblance définie par :

$$\frac{d\Lambda_Z(x)}{dx} = \frac{dp(Z|x)}{dx} = 0 \quad (2.8)$$

Estimateur du maximum *a posteriori*

L'estimateur équivalent à l'estimateur ML, lorsque le paramètre à estimer est aléatoire, correspond à l'*estimateur du maximum a posteriori (maximum a posteriori (MAP) estimator)* :

$$\hat{x}^{\text{MAP}}(Z) = \arg \max_x p(x|Z) = \arg \max_x [p(Z|x)p(x)] \quad (2.9)$$

Cette expression découle de la formule de Bayes décrite dans (2.5). La constante de normalisation n'a aucune influence sur la maximisation.

Cas gaussien

Considérons une mesure portant sur un paramètre inconnu x en présence d'un bruit w :

$$z = x + w \quad (2.10)$$

Le bruit possède une distribution gaussienne centrée sur zéro et de variance σ^2 :

$$w \sim \mathcal{N}(0, \sigma^2) \quad (2.11)$$

Supposons d'abord que x est une constante inconnue sur laquelle on n'a aucune connaissance préalable. La fonction de vraisemblance de x est (sans indices pour simplifier) :

$$\Lambda(x) = p(z|x) = \mathcal{N}(z|x, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(z-x)^2}{2\sigma^2}} \quad (2.12)$$

L'estimateur à maximum de vraisemblance est donc :

$$\hat{x}^{\text{ML}} = \arg \max_x \Lambda(x) = z \quad (2.13)$$

car le maximum de (2.12) se produit lorsque $x = z$.

Si l'on considère maintenant que notre connaissance *a priori* sur x suit une loi normale de moyenne \bar{x} et de variance σ_0^2 , soit :

$$p(x) = \mathcal{N}(x|\bar{x}, \sigma_0^2) \quad (2.14)$$

et que x est indépendant du bruit w , alors la densité de probabilité *a posteriori* de x conditionnée par l'observation z est :

$$p(x|z) = \frac{p(z|x)p(x)}{p(z)} = \frac{1}{c} e^{-\frac{(z-x)^2}{2\sigma^2} - \frac{(x-\bar{x})^2}{2\sigma_0^2}} \quad (2.15)$$

c étant une constante de normalisation indépendante de x :

$$c = 2\pi\sigma\sigma_0 p(z) \quad (2.16)$$

On peut facilement démontrer que la formule (2.15) correspond à la densité de probabilité

d'une loi gaussienne [9] :

$$p(x|z) = \mathcal{N}[x|\xi(z), \sigma_1^2] = \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{[x-\xi(z)]^2}{2\sigma_1^2}} \quad (2.17)$$

ayant pour moyenne :

$$\xi(z) = \frac{\sigma^2}{\sigma_0^2 + \sigma^2} \bar{x} + \frac{\sigma_0^2}{\sigma_0^2 + \sigma^2} z \quad (2.18)$$

et pour variance :

$$\sigma_1^2 = \frac{\sigma_0^2 \sigma^2}{\sigma_0^2 + \sigma^2} \quad (2.19)$$

La maximisation de (2.17) implique immédiatement :

$$\hat{x}^{\text{MAP}} = \xi(z) \quad (2.20)$$

Notons ici que dans le cas gaussien, l'estimateur MAP donné par (2.18) est une somme pondérée de :

- z , l'estimateur ML, et en même temps la mesure, qui maximise la fonction de vraisemblance
- \bar{x} , qui maximise la densité de probabilité *a priori* du paramètre à estimer

L'équation (2.18) peut aussi être écrite de la manière suivante :

$$\hat{x}^{\text{MAP}} = \xi(z) = \frac{1}{\sigma_0^{-2} + \sigma^{-2}} \left(\frac{\bar{x}}{\sigma_0^2} + \frac{z}{\sigma^2} \right) \quad (2.21)$$

Les coefficients de pondération des éléments de cette somme z et \bar{x} sont inversement proportionnels à leurs variances, σ^2 et σ_0^2 respectivement. L'estimateur donne plus de poids à l'élément le plus certain : la mesure ou connaissance *a priori*. C'est ce même principe qui est utilisé dans le filtre de Kalman abordé ultérieurement.

2.2.2 Estimateurs des moindres carrés et au sens des moindres carrés

Estimateur des moindres carrés

Une autre méthode d'estimation courante pour les paramètres non aléatoires concerne la *méthode des moindres carrés (least squares (LS))*.

En considérant les mesures scalaires et non linéaires :

$$z_j = h(j, x) + w_j \quad j = 1, \dots, k \quad (2.22)$$

l'estimateur des moindres carrés de x est :

$$\hat{x}^{\text{LS}}(k) = \arg \min_x \left(\sum_{j=1}^k [z_j - h(j, x)]^2 \right) \quad (2.23)$$

Estimateur au sens des moindres carrés

Pour les paramètres aléatoires, l'équivalent de l'estimateur LS est *l'estimateur au sens des moindres carrés (minimum mean square error (MMSE) estimator)* :

$$\hat{x}^{\text{MMSE}}(Z) = \arg \min_{\hat{x}} E[(\hat{x} - x)^2 | Z] \quad (2.24)$$

La solution est calculée par annulation de la dérivée de (2.24) :

$$\frac{d}{d\hat{x}} E[(\hat{x} - x)^2 | Z] = E[2(\hat{x} - x) | Z] = 2(\hat{x} - E(x | Z)) = 0 \quad (2.25)$$

L'estimateur MMSE s'écrit alors :

$$\hat{x}^{\text{MMSE}}(Z) = E(x | Z) \triangleq \int_{-\infty}^{+\infty} xp(x | Z) dx \quad (2.26)$$

2.2.3 Le filtre de Kalman

Le filtre de Kalman est un ensemble d'équations qui fournit une méthode récursive pour estimer l'état d'un processus de façon à minimiser la moyenne du carré de l'erreur. Le filtre de Kalman est puissant sur plusieurs aspects : il supporte l'estimation des états passés, présents et même futurs, et ceci est valable même si le vrai modèle du système est inconnu.

Depuis la publication en 1960 par R.E. Kalman d'un article décrivant une solution récursive au problème de filtrage linéaire des données discrètes [37], le filtre de Kalman a été le sujet de recherches intensives et d'applications dans différents domaines.

Principe du filtre de Kalman

Le filtre de Kalman traite le problème général de l'estimation de l'état $x \in \mathbb{R}^n$ d'un système discret régi par l'équation aux différences suivante :

$$x_k = Fx_{k-1} + v_{k-1} \quad (2.27)$$

avec des mesures (ou observations) $z \in \mathbb{R}^m$ telles que :

$$z_k = Hx_k + w_k \quad (2.28)$$

Dans l'équation aux différences (2.27), la matrice F , de dimension $n \times n$, représente la transition entre l'état à l'instant $k-1$ et l'instant k . Dans l'équation de mesure (2.28), la matrice H , de dimension $m \times n$, relie l'état x aux mesures z . Les matrices F et H peuvent varier dans le temps (elles sont alors notées F_k et H_k respectivement), mais on les considère constantes dans la suite. L'état initial x_0 est supposé connu et ayant une distribution gaussienne.

v_k et w_k dénotent les bruits du processus et des mesures respectivement et sont supposés indépendants, blancs, centrés et gaussiens :

$$v \sim \mathcal{N}(0, Q) \quad \text{avec} \quad Q = E(v_k v_k^T) \quad (2.29)$$

$$w \sim \mathcal{N}(0, R) \quad \text{avec} \quad R = E(w_k w_k^T) \quad (2.30)$$

Les matrices de covariance du bruit du processus Q et du bruit des mesures R peuvent changer à chaque pas de mesure, ici on suppose qu'elles sont constantes.

La linéarité des équations d'état (2.27) et de mesure (2.28), et la nature gaussienne de l'état initial x_0 ainsi que des bruits d'état v et de mesure w permet de préserver l'hypothèse *linéaire-gaussienne* du filtre de Kalman le long du processus de filtrage.

Origine probabiliste du filtre de Kalman

Le filtre de Kalman repose sur l'inférence bayésienne décrite par la formule de Bayes abordée dans la section 2.2 :

$$p(x|z) = \frac{p(z|x)p(x)}{p(z)} \quad (2.31)$$

L'objectif est d'estimer la distribution *a posteriori* $p(x|z)$ de l'état du système x connaissant les mesures z . Par application de la formule de Bayes, cette estimation est calculée à partir de :

- La vraisemblance $p(z|x)$: qui mesure la proximité de l'état à la mesure
- La loi *a priori* $p(x)$: qui donne une mesure de confiance sur la valeur de l'état, fournie par le modèle
- La distribution des mesures $p(z)$: inconnue en pratique

Ces distributions sont calculées comme suit :

- La loi *a priori*, qui définit le modèle d'évolution de l'état, possède une structure markovienne, autrement dit, l'état x_{k+1} ne dépend que de l'état précédent, d'où :

$$f(x_{k+1}|x_k) = p(x_{k+1}|x_{0:k}) = \mathcal{N}(x_{k+1}|Fx_k, Q) \quad (2.32)$$

$x_{0:k}$ dénote tous les vecteurs d'état entre les instants 0 et k .

- La distribution de vraisemblance : on suppose que les états et les mesures sont indépendants et identiquement distribués :

$$g(z_k|x_k) = p(z_k|x_{0:k}, z_{0:k-1}) = \mathcal{N}(z_k|Hx_k, R) \quad (2.33)$$

- On suppose connue la distribution statistique de l'état initial :

$$p(x_0) = \mathcal{N}(x_0|\bar{x}_0, P_0) \quad (2.34)$$

Ainsi, la distribution *a posteriori* décrite par l'équation (2.31) peut s'écrire sous la forme :

$$p(x_{k+1}|z_{0:k+1}) = \frac{g(z_{k+1}|x_{k+1}) p(x_{k+1}|z_{0:k})}{p(z_{k+1}|z_{0:k})} \quad (2.35)$$

Etape de prédiction (loi *a priori*)

La loi *a priori* constitue la première étape du filtre de Kalman : la *prédiction*. L'équation de Chapman-Kolmogorov définit l'état prédit en fonction des observations précédentes :

$$p(x_{k+1}|z_{0:k}) = \int f(x_{k+1}|x_k) p(x_k|z_{0:k}) dx_k \quad (2.36)$$

La notation suivante sera utilisée pour représenter la moyenne conditionnelle :

$$\hat{x}_{j|k} \triangleq E(x_j|Z_k) \quad \text{avec} \quad Z_k \triangleq \{z_j\}_{j=1}^k \quad (2.37)$$

On suppose que la distribution de l'état à l'instant k , définie par l'ensemble des mesures jusqu'à l'instant k , est gaussienne de moyenne $\hat{x}_{k|k}$ et de matrice de covariance $P_{k|k}$:

$$p(x_k|z_{0:k}) = \mathcal{N}(x_k|\hat{x}_{k|k}, P_{k|k}) \quad (2.38)$$

A partir de l'équation d'état du système (2.27), l'état prédit à l'instant $k+1$ possède aussi une distribution gaussienne :

$$p(x_{k+1}|z_{0:k}) = \mathcal{N}(x_{k+1}|\hat{x}_{k+1|k}, P_{k+1|k}) \quad (2.39)$$

Avec :

$$\hat{x}_{k+1|k} = F\hat{x}_{k|k} \quad (2.40)$$

$$P_{k+1|k} = Q + FP_{k|k}F^T \quad (2.41)$$

Q traduit l'incertitude sur le modèle et $FP_{k|k}F^T$ la covariance de l'état estimé.

Etape de correction (loi *a posteriori*)

A partir de la distribution *a posteriori* décrite par (2.35) et en remplaçant la distribution de la vraisemblance et de la loi *a priori* par les expressions trouvées dans les formules (2.33) et (2.39), on a :

$$p(x_{k+1}|z_{0:k+1}) \propto \mathcal{N}(z_{k+1}|Hx_{k+1}, R) \mathcal{N}(x_{k+1}|\hat{x}_{k+1|k}, P_{k+1|k}) \quad (2.42)$$

$$\begin{aligned} &\propto \exp\left(-\frac{1}{2}[(z_{k+1} - Hx_{k+1})^T R^{-1}(z_{k+1} - Hx_{k+1})]\right) \\ &\quad \times \exp\left(-\frac{1}{2}[(x_{k+1} - \hat{x}_{k+1|k})^T P_{k+1|k}^{-1}(x_{k+1} - \hat{x}_{k+1|k})]\right) \end{aligned} \quad (2.43)$$

Cette distribution suit une loi gaussienne de moyenne $\hat{x}_{k+1|k+1}$ et de matrice de covariance P_{k+1} :

$$p(x_{k+1}|z_{0:k+1}) \propto \exp\left(-\frac{1}{2}[(x_{k+1} - \hat{x}_{k+1|k+1})^T P_{k+1}^{-1}(x_{k+1} - \hat{x}_{k+1|k+1})]\right) \quad (2.44)$$

$$= \mathcal{N}(x_{k+1}|\hat{x}_{k+1|k+1}, P_{k+1}) \quad (2.45)$$

Par identification des équations (2.44) et (2.45), on démontre que :

$$\hat{x}_{k+1|k+1} = P_{k+1} \left(H^T R^{-1} z_{k+1} + P_{k+1}^{-1} \hat{x}_{k+1|k} \right) \quad (2.46)$$

$$P_{k+1} = (H^T R^{-1} H + P_{k+1}^{-1})^{-1} \quad (2.47)$$

Ces expressions peuvent aussi être réécrites sous la forme :

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_k(z_{k+1} - H\hat{x}_{k+1|k}) \quad (2.48)$$

$$K_k = P_{k+1|k}H^T(H P_{k+1|k}H^T + R)^{-1} \quad (2.49)$$

$$P_{k+1} = (I - K_kH)P_{k+1|k} \quad (2.50)$$

K_k est appelé *gain du filtre de Kalman*, il traduit le degré de confiance que l'on donne à la mesure ou au modèle (la prédiction). Ainsi, dans l'équation (2.48), si le gain K_k est faible, la moyenne de la loi *a posteriori* $\hat{x}_{k+1|k+1}$ tend vers la moyenne de la prédiction $\hat{x}_{k+1|k}$. En revanche, si le gain K_k est élevé, alors $\hat{x}_{k+1|k+1}$ est mis à jour par l'innovation (ou le *résidu*) $z_{k+1} - H\hat{x}_{k+1|k}$ qui représente la variation apportée par les observations.

Le terme $H\hat{x}_{k+1|k}$ est appelé la *prédiction de mesure*, elle représente la mesure devant correspondre à la prédiction $\hat{x}_{k+1|k}$, elle est notée $\hat{z}_{k+1|k}$:

$$\hat{z}_{k+1|k} = H\hat{x}_{k+1|k} \quad (2.51)$$

L'innovation, souvent notée ν_{k+1} ou \tilde{z}_{k+1} , est la différence entre la mesure et la prédiction de mesure $\hat{z}_{k+1|k}$:

$$\nu_{k+1} = z_{k+1} - \hat{z}_{k+1|k} = z_{k+1} - H\hat{x}_{k+1|k} \quad (2.52)$$

Sa matrice de covariance S_{k+1} est calculée de la même manière que (2.41) :

$$S_{k+1} = R + H P_{k+1|k} H^T \quad (2.53)$$

Algorithme du filtre de Kalman

Le filtre de Kalman peut être programmé de façon récursive en deux étapes : l'*étape de prédiction* qui utilise le modèle pour calculer une prédiction de l'état actuel à partir de l'estimée de l'état à l'instant précédent, et l'*étape de correction* ou de *mise à jour* où l'on utilise les mesures pour calculer l'estimation de l'état au moment présent. Cette étape est une combinaison de la prédiction et de la mesure, où les facteurs de pondération dépendent des matrices des covariances de chacune des combinaisons.

Ces étapes sont résumées sur la table 2.1.

Initialisation	
Estimée initiale	$\hat{x}_{0 0} = E(X_0)$
Prédiction	
Prédiction de l'état	$\hat{x}_{k+1 k} = F \hat{x}_{k k}$
Covariance de l'erreur de prédiction	$P_{k+1 k} = Q + F P_{k k} F^T$
Correction	
Estimée	$\hat{x}_{k+1 k+1} = \hat{x}_{k+1 k} + K_k (z_{k+1} - H \hat{x}_{k+1 k})$
Covariance de l'estimée	$P_{k+1} = (I - K_k H) P_{k+1 k}$
Gain du filtre	$K_k = P_{k+1 k} H^T (H P_{k+1 k} H^T + R)^{-1}$

TABLE 2.1 – Algorithme du filtre de Kalman

2.2.4 Méthodes de Monte Carlo : le filtre particulaire

La technique de Monte Carlo est une méthode d'échantillonnage stochastique utilisée dans l'analyse de systèmes complexes difficiles à traiter analytiquement, et plus particulièrement le calcul d'intégrales numériques. Durant les dernières années, les approches Monte Carlo séquentielles ont attiré l'attention des chercheurs dans différents domaines tels que : le traitement du signal, l'automatique, la biologie, les télécommunications, ... [25]. L'un des mérites de ces méthodes est qu'elles permettent une estimation en ligne en combinant la puissance de l'échantillonnage de Monte Carlo et les méthodes d'inférence bayésienne avec un coût de calcul raisonnable.

Dans le domaine du filtrage statistique, on a souvent besoin d'évaluer l'espérance d'une fonction d'état h telle que :

$$\bar{h} = E(h(x_k)) = \int h(x_k) p(x_k | z_{0:k}) dx_k \quad (2.54)$$

Lorsque cette intégrale ne peut pas être calculée analytiquement, on peut l'approximer par l'intégration de Monte Carlo :

$$\hat{h} = \frac{1}{N} \sum_{i=1}^N h(x_k^{(i)}) \quad (2.55)$$

Où $x_k^{(i)}$ sont N particules tirés aléatoirement avec la loi :

$$x_k^{(i)} \stackrel{iid}{\sim} p(x_k | z_{0:k}) \quad (2.56)$$

En pratique, la distribution $p(x_k|z_{0:k})$ n'est pas connue, on génère alors les particules en utilisant une distribution quelconque q dite *fonction d'importance* :

$$x_k^{(i)} \sim q(x_k) \quad , \quad i = 1, \dots, N \quad (2.57)$$

On affecte un poids $w_k^{(i)}$ pour chaque particule de la manière suivante :

$$w_k^{(i)} \propto \frac{p(x_k^{(i)}|z_{0:k})}{q(x_k^{(i)})} \quad (2.58)$$

Si on normalise ces poids tel que $\sum_{i=1}^N w_k^{(i)} = 1$, on peut former une approximation de la distribution $p(x_k|z_{0:k})$ en utilisant la distribution de Dirac $\delta(\cdot)$:

$$p(x_k|z_{0:k}) \approx \sum_{i=1}^N w_k^{(i)} \delta(x_k - x_k^{(i)}) \quad (2.59)$$

La figure 2.1 illustre graphiquement la répartition des échantillons pris selon la distribution connue $q(x)$ et ceux qui devraient être pris selon la distribution inconnue $p(x)$. La densité des échantillons est d'autant plus importante que la probabilité est élevée.

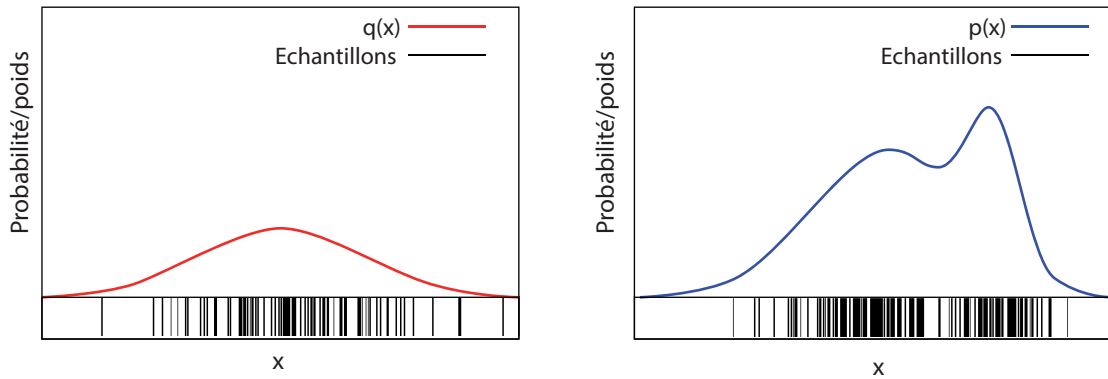


FIGURE 2.1 – Échantillonnage de Monte Carlo selon [75]

La figure 2.2 montre visuellement les particules prises selon la distribution $q(x)$ et leur poids calculé par la formule 2.58 donnant naissance à l'approximation 2.59.

Ainsi, l'estimation de l'espérance dans 2.55 peut être réécrite comme suit :

$$\hat{h} = \sum_{i=1}^N w_k^{(i)} h(x_k^{(i)}) \quad (2.60)$$

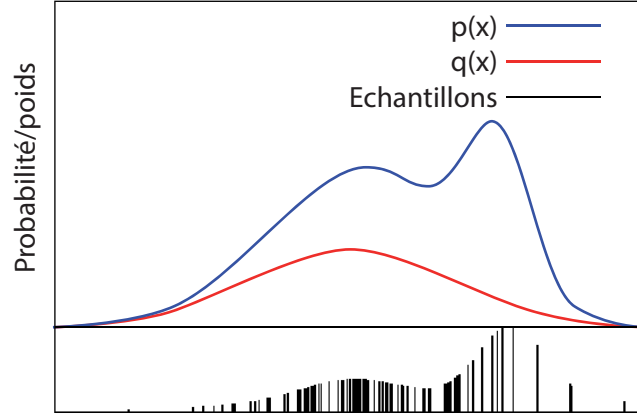


FIGURE 2.2 – Poids des échantillons selon [75]

$$\text{où : } x_k^{(i)} \stackrel{iid}{\sim} q(x_k) , \quad w_k^{(i)} \propto \frac{p(x_k^{(i)} | z_{0:k})}{q(x_k^{(i)})} , \quad \sum_{i=1}^N w_k^{(i)} = 1$$

Et par la suite, l'approximation de l'intégrale 2.54 devient :

$$\bar{h} = E(h(x_k)) \approx \int h(x_k) \sum_{i=1}^N w_k^{(i)} \delta(x_k - x_k^{(i)}) dx_k = \sum_{i=1}^N w_k^{(i)} h(x_k^{(i)}) \quad (2.61)$$

Principe du filtre particulaire

Le filtre particulaire repose sur la méthode Monte Carlo et le filtrage bayésien pour trouver $p(x_k^{(i)} | z_{0:k})$ de façon récursive. A l'instar du filtre de Kalman, le filtre se déroule en deux étapes : la prédiction et la correction avec une étape optionnelle de ré-échantillonnage.

Etape de prédiction

Soit N particules distribuées suivant la loi $p(x_k | z_{0:k})$ notées :

$$x_k^{(i)} , \quad i = 1, 2, \dots, N \quad \text{avec } N \gg 1 \quad (2.62)$$

On génère de nouvelles particules suivant la distribution de transition :

$$x_{k+1}^{(i)} \sim f(x_{k+1} | x_k^{(i)}) \quad (2.63)$$

Chaque paire $(x_k^{(i)}, x_{k+1}^{(i)})$ suit la distribution jointe $p(x_k, x_{k+1} | z_{0:k})$. De plus, $x_{k+1}^{(i)}$ considérée séparément est un échantillon aléatoire de la distribution marginale $p(x_{k+1} | z_{0:k})$.

Etape de correction

La distribution *a posteriori* est obtenue par la loi de Bayes :

$$p(x_{k+1}|z_{0:k+1}) = \frac{g(z_{k+1}|x_{k+1}) p(x_{k+1}|z_{0:k})}{p(z_{k+1}|z_{0:k})} \quad (2.64)$$

Dans cette étape de correction, on approxime la distribution *a posteriori* en calculant les poids :

$$w_{k+1} \propto \frac{p(x_{k+1}|z_{0:k+1})}{q(x_{k+1})} \quad (2.65)$$

Dans le cas général, la loi $q(\cdot)$ peut être quelconque, mais ici les particules sont générées dans l'étape de prédiction suivant la loi $p(x_{k+1}|z_{0:k})$, d'où, et en utilisant l'équation 2.64 :

$$w_{k+1} \propto \frac{\frac{g(z_{k+1}|x_{k+1}) p(x_{k+1}|z_{0:k})}{p(z_{k+1}|z_{0:k})}}{p(x_{k+1}|z_{0:k})} \quad (2.66)$$

Sachant que la probabilité $p(z_{k+1}|z_{0:k})$ est constante :

$$w_{k+1} \propto g(z_{k+1}|x_{k+1}) \quad (2.67)$$

Ainsi, les poids sont distribués suivant la distribution de vraisemblance des particules. En pratique, les poids sont normalisés de la manière suivante :

$$w_{k+1}^{(i)} = \frac{\hat{w}_{k+1}^{(i)}}{\sum_{j=1}^N \hat{w}_{k+1}^{(j)}} \quad \text{avec} \quad \hat{w}_{k+1}^{(i)} = g(z_{k+1}|x_{k+1}^{(i)}) \quad (2.68)$$

Les poids sont accumulés avec le temps afin de donner plus d'importance aux particules les plus probables :

$$w_{k+1} \propto w_k g(z_{k+1}|x_{k+1}) \quad (2.69)$$

L'état est estimé par :

$$\hat{x}_{k+1} = \sum_{i=1}^N \hat{w}_{k+1}^{(i)} x_{k+1}^{(i)} \quad (2.70)$$

Etape de ré-échantillonnage

Un problème courant dans le filtre particulaire est le phénomène de dégénérescence, où, après quelques itérations, toutes les particules sauf une ont un poids négligeable. De ce fait, un temps de calcul important devient associé à la mise à jour des particules dont la contribution à l'approximation de loi *a posteriori* est quasiment nulle. Une mesure de la dégénérescence a été proposée dans [14, 58] par le calcul de la taille effective des échantillons N_{eff} :

$$N_{eff} = \frac{N}{1 + \text{var}(w_{k+1}^{*(i)})} \quad (2.71)$$

Où $w_{k+1}^{*(i)} = p(x_{k+1}^{(i)} | z_{0:t}) / q(x_{k+1}^{(i)} | x_k^{(i)}, z_k)$ est appelé le *vrai poids*. Ceci ne peut pas être évalué exactement, mais une estimation $\widehat{N_{eff}}$ peut être obtenue par :

$$\widehat{N_{eff}} = \frac{1}{\sum_{i=1}^N (w_{k+1}^{(i)})^2} \quad (2.72)$$

Un faible N_{eff} indique une dégénérescence sévère. En pratique, on fixe un seuil N_T pour détecter la dégénérescence, et lorsque N_{eff} devient inférieur à ce seuil on ré-échantillonne les particules.

L'idée d'origine concernant l'étape de ré-échantillonnage consiste à éliminer les particules ayant un poids faible et de se concentrer sur les particules ayant un poids important. Le ré-échantillonnage implique la génération d'un nouvel ensemble de particules $x_{k+1}^{*(i)}$ ayant des probabilités identiques : $w_{k+1}^{*(i)} = 1/N$.

Il existe plusieurs algorithmes de ré-échantillonnage, les plus connus sont les ré-échantillonnages multinomial, stratifié et systématique [31].

2.3 Principe de l'approche multi-modèles

La poursuite de cibles (*Target tracking*) est l'une des applications les plus répandues où l'on essaye d'estimer l'état de la trajectoire d'un objet en mouvement ou *cible*. Cette cible est souvent traitée comme un objet ponctuel sans forme dont on essaye d'estimer la dynamique. Ce modèle dynamique, ou modèle de mouvement, décrit l'évolution de l'état de la cible x en fonction du temps. La cible obéit à un modèle de mouvement pendant une certaine durée, mais

peut effectuer des manœuvres à des instants aléatoires. La plupart des méthodes de poursuite des cibles manœuvrantes reposent sur l'analyse des modèles. Ces techniques supposent que le mouvement de la cible ainsi que les observations peuvent être représentés par un certain nombre de modèles mathématiques bien définis [48]. Chacun de ces modèles correspond à un *mode* de fonctionnement différent du système (par exemple : mouvement avec vitesse constante, mouvement accéléré, mouvement décéléré, rotation à gauche ou à droite, ...). Pour de tels systèmes, il s'agit de faire une estimation dite *hybride* car le processus observé possède deux composantes de natures différentes : une composante continue qui est l'état du système et une composante discrète qui est le *mode* ou le *régime opérationnel* du système.

La différence entre un mode et un modèle indique que le mode se réfère à un comportement ou à une structure du système, tandis que le modèle est une représentation mathématique ou une description de ce comportement avec un certain degré de précision [53]. L'ensemble des modèles \mathbb{M} se différencie par rapport à l'espace des modes \mathbb{S} en deux aspects : 1) leur nombre d'éléments est différent : \mathbb{M} a souvent moins d'éléments que \mathbb{S} , 2) un modèle est souvent une description simplifiée du mode. Par exemple, on peut utiliser un ensemble réduit de modèles : un modèle de mouvement sans manœuvre en plus d'un certain nombre de modèles de rotation à vitesse constante dans la poursuite d'une cible qui peut dans la pratique effectuer plusieurs manœuvres complexes (modes).

L'idée dans la poursuite de cibles est de décider à chaque moment quel est le modèle qui représente au mieux le système, puis d'exécuter un filtre reposant sur ce modèle comme s'il s'agissait du modèle réel du système. Il existe plusieurs méthodes reposant sur l'approche décisionnelle. Elles sont décrites dans [52], cependant cette approche souffre de certains défauts dont on cite [53] : il y a toujours des erreurs dans le choix du bon modèle, et en plus, la décision est prise d'une façon irrévocable avant l'estimation, alors que l'estimation apporte fréquemment des informations supplémentaires qui nous aident souvent à définir le bon modèle. L'approche MM contourne cette problématique en utilisant plusieurs modèles. Son principe est de considérer un ensemble \mathbb{M} de modèles candidats pour représenter le mode réel à l'instant considéré. Elle exécute un banc de filtres élémentaires, chacun reposant sur un modèle, puis génère l'estimation finale à partir des résultats des filtres.

L'estimation multi-modèles repose sur quatre composants clés illustrés sur la figure 2.3 pour deux modèles :

1. *Détermination de l'ensemble des modèles* : Ceci inclut les techniques de conception *off-line* et les techniques *on-line* qui adaptent l'ensemble des modèles en fonction des ob-

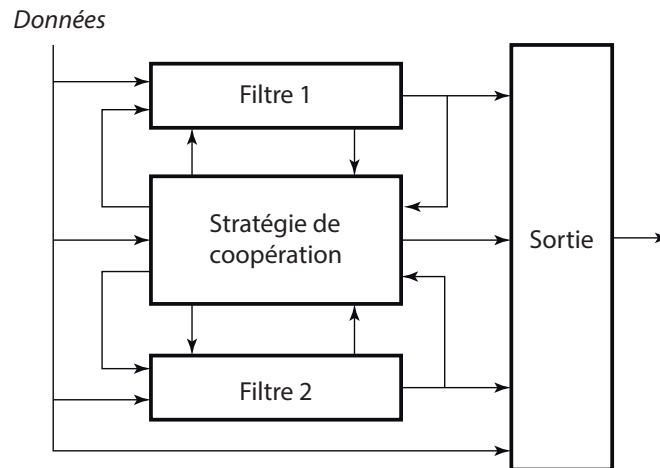


FIGURE 2.3 – Structure générale de l'estimation multi-modèles selon [53]

servations. La performance d'un estimateur MM dépend largement de l'ensemble des modèles utilisés.

2. *La stratégie de coopération* : Elle se réfère aux méthodes d'estimation des incertitudes relatives à l'ensemble des modèles et à la séquence de modèles : choix des modèles à écarter, mixage des séquences similaires et choix de la meilleure séquence de modèles à considérer.
3. *Filtrage conditionnel* : Il s'agit de l'estimation récursive des composantes du système (l'état), conditionnée par la séquence de modes. On utilise généralement les méthodes d'estimation conventionnelle de l'état comme le filtre de Kalman.
4. *Traitement de la sortie* : Il s'agit du processus de génération de l'estimée finale en rapportant aux résultats des filtres et aux mesures. Elle inclut la fusion ou la combinaison des différentes estimées obtenues par les filtres et le choix de la meilleure d'entre elles.

Trois générations d'algorithmes MM ont été développées. La première génération : l'estimation MM autonome (AMM) a été proposée par Magill [60] et Lainiotis [41] et elle a été largement utilisée par Maybeck [62]. La seconde génération : l'estimation MM coopérative (CMM), qui comprend l'algorithme IMM développé par Blom [20], a connu un grand succès grâce aux différentes applications de cet algorithme dans la poursuite des cibles. La popularité de ce filtre est due aux livres sur l'estimation dynamique de Bar-Shalom (par exemple : [8, 9]) et aux travaux de Blom [20, 21, 22] et Blair [19, 18]. La troisième génération, l'estimation MM à structure variable (VSMM), caractérisée par sa structure à modèles variable, gagne de l'élan actuellement. X.R. Li et son équipe ont proposé cette approche et son utilisation dans plusieurs applications (voir par exemple : [45, 48, 50]). Les estimateurs MM reposent sur une approche

de décision dite *douce* ou *souple* (*soft decision*). La génération de l'estimée finale est une fusion d'informations entre les différentes estimées fournies par les filtres élémentaires.

Il existe une seconde catégorie d'estimateurs MM qui utilise une approche de décision dite *dure* (*Hard decision*) qui identifie un "bon" sous-ensemble \mathbb{B} de séquences de modèles par une procédure de décision dure, puis génère l'estimée finale à partir des estimées conditionnées par ces séquences de modèles. L'ensemble \mathbb{B} peut changer dans le temps ou lorsque plus d'informations sont collectées. Plusieurs algorithmes de décision ont été proposés dans ce sens. Les plus connus sont les algorithmes *B-Best* [85] et *Viterbi* [5]. Ces algorithmes appliquent la technique d'élagage (*pruning*) qui consiste à écarter les séquences de modèles les plus invraisemblables. D'autres techniques de prise de décision ont été utilisées comme les systèmes experts, les réseaux de neurones, les algorithmes heuristiques, les algorithmes génétiques, la logique floue et le raisonnement par l'évidence. Cette catégorie de filtre MM que nous n'utilisons pas dans notre problématique ne sera pas décrite dans cette thèse.

2.4 L'estimateur multi-modèles statique ou autonome

La première génération d'estimateurs MM est l'estimateur MM statique appelé aussi algorithme multi-modèles autonome (*AMM : Autonomus MM*), dans lequel le modèle auquel le système obéit est fixe et il n'y a pas de transition vers un autre modèle pendant le processus d'estimation. Bien que le modèle effectif soit fixe, les autres modèles continuent à évoluer selon leur propre dynamique.

Le modèle qui est supposé actif pendant le processus d'estimation est un parmi r modèles (ou *modes*) possibles :

$$M \in \mathbb{M} \quad \text{tel que} \quad \mathbb{M} = \{M^j\}_{j=1}^r \quad (2.73)$$

On note la probabilité *a priori* que le modèle M^j soit correct (que le système suit le modèle j) par :

$$P(M^j|Z_0) = \mu_0^j \quad j = 1, \dots, r \quad (2.74)$$

Où Z_0 est l'information *a priori* et :

$$\sum_{j=1}^r \mu_0^j = 1 \quad (2.75)$$

puisque le modèle correct est parmi les r modèles possibles.

On suppose dans la suite que tous les modèles sont linéaires et gaussiens. Mais ceci n'empêche pas l'utilisation de cette approche avec les systèmes non-linéaires si on procède à leur linéarisation autour du point estimé.

L'estimateur MM autonome est construit sur la base de r filtres de Kalman associés à des modèles fixes indépendants. En appliquant la formule de Bayes (2.5), la probabilité *a posteriori* que le modèle j soit correct en connaissant les mesures jusqu'à l'instant k est donnée par :

$$\begin{aligned}\mu_k^j &\triangleq P(M^j|Z_k) = P(M^j|z_k, Z_{k-1}) \\ &= \frac{p(z_k|Z_{k-1}, M^j)P(M^j|Z_{k-1})}{p(z_k|Z_{k-1})} \\ &= \frac{p(z_k|Z_{k-1}, M^j)P(M^j|Z_{k-1})}{\sum_{i=1}^r p(z_k|Z_{k-1}, M^i)P(M^i|Z_{k-1})}\end{aligned}\quad (2.76)$$

Ou encore :

$$\mu_k^j = \frac{p(z_k|Z_{k-1}, M^j)\mu_{k-1}^j}{\sum_{i=1}^r p(z_k|Z_{k-1}, M^i)\mu_{k-1}^i} \quad \text{pour } j = 1, \dots, r \quad (2.77)$$

Ceci permet de calculer d'une façon récursive la probabilité de chaque modèle à partir de la probabilité *a priori* donnée par (2.74).

Le premier terme du numérateur de l'équation (2.77) est la fonction de vraisemblance du mode j à l'instant k , son espérance s'écrit :

$$E(z_k|Z_{k-1}, M^j) = E(Hx_{k|k} + w_k|Z_{k-1}, M^j) \quad (2.78)$$

Etant donné que le bruit de mesure w_k est blanc et de moyenne nulle, cette espérance devient :

$$E(z_k|Z_{k-1}, M^j) = \hat{z}_{k|k-1}^j = H\hat{x}_{k|k-1}^j \quad (2.79)$$

Sa covariance est la même que celle de l'innovation du filtre j décrite par l'équation (2.53) :

$$S_k^j = R + HP_{k|k-1}^j H^T \quad (2.80)$$

Etant donné que les modèles sont supposés linéaires et gaussiens, cette fonction de vraisem-

blance pour le filtre j qu'on note Λ_k^j s'écrit :

$$\begin{aligned}\Lambda_k^j &\triangleq p(z_k | Z_{k-1}, M^j) = \mathcal{N}(z_k | \hat{z}_{k|k-1}^j, S_k^j) = \mathcal{N}(z_k - \hat{z}_{k|k-1}^j | 0, S_k^j) \\ &= \mathcal{N}(\nu_k^j | 0, S_k^j)\end{aligned}\quad (2.81)$$

ν_k^j et S_k^j sont respectivement l'innovation et la covariance de l'innovation du filtre associé au modèle j .

L'équation (2.77) devient :

$$\mu_k^j = \frac{\Lambda_k^j \mu_{k-1}^j}{\sum_{i=1}^r \Lambda_k^i \mu_{k-1}^i} \quad \text{pour } j = 1, \dots, r \quad (2.82)$$

Ainsi dans l'estimateur multi-modèles statique, chaque filtre de Kalman est associé à un modèle qui évolue indépendamment des autres. La probabilité que l'un de ces modèles soit correct est calculée par l'équation (2.82) en se reportant à la fonction de vraisemblance (2.81). Le modèle le plus probable est celui dont l'estimée reste la plus proche des observations. Cette structure est illustrée par la figure 2.4.

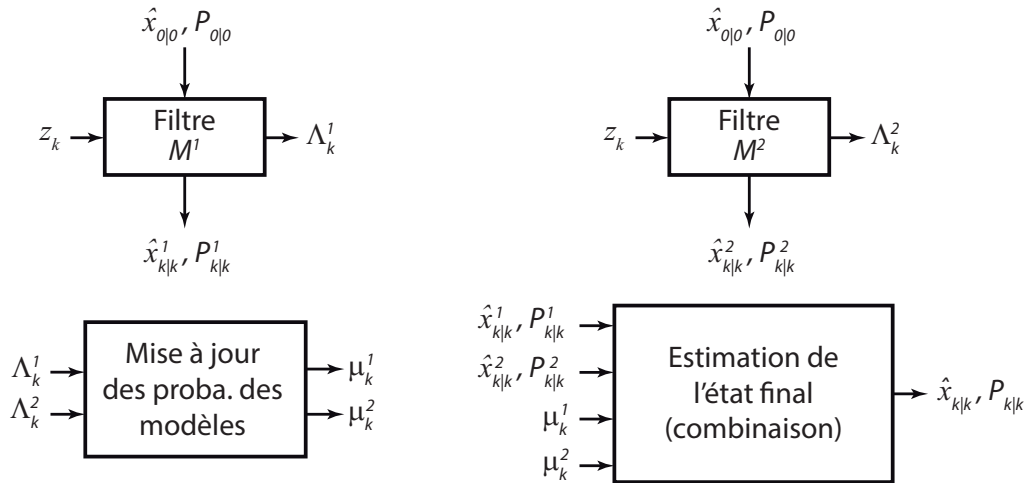


FIGURE 2.4 – Structure de l'estimateur multi-modèles statique selon [9]

L'estimée de chaque filtre j à l'instant k est notée \hat{x}_k^j , la covariance correspondante P_k^j et sa fonction de vraisemblance Λ_k^j . On représente sur la figure 2.5 le cycle de fonctionnement de l'estimateur. Dans cette représentation la combinaison des estimés est définie par un opérateur de fusion.

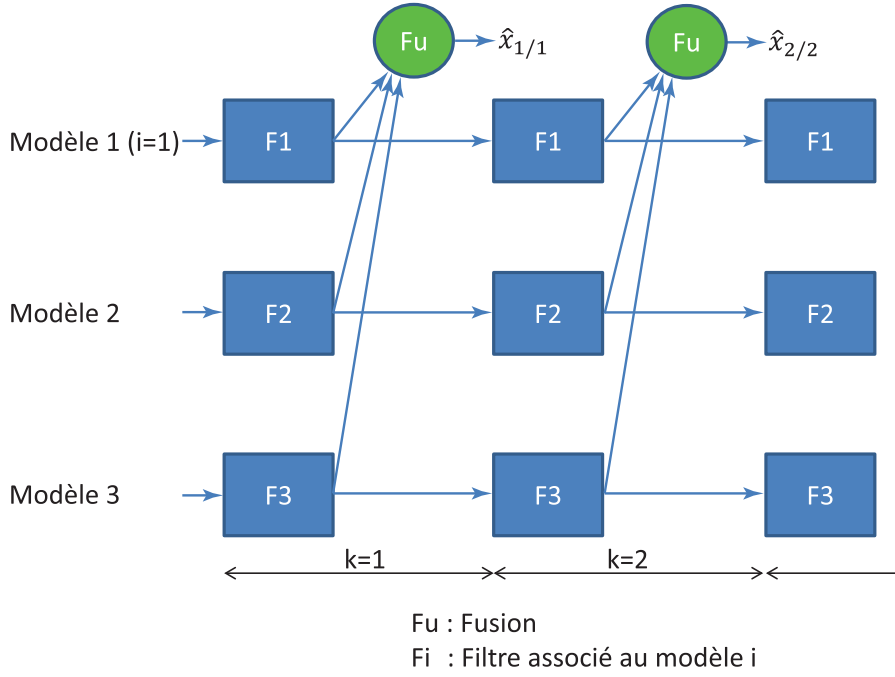


FIGURE 2.5 – Cycle de l'estimateur multi-modèles autonome

L'estimée finale est une somme pondérée des estimées de chaque filtre ayant pour facteur de pondération la probabilité du modèle associé. Si on suppose que les états des modèles sont gaussiens, la densité de probabilité de l'état du système est alors une combinaison de r lois gaussiennes :

$$p(x_k | Z_k) = \sum_{j=1}^r \mu_k^j \mathcal{N}(x_k | \hat{x}_{k|k}^j, P_{k|k}^j) \quad (2.83)$$

ayant pour moyenne et covariance [9] :

$$\hat{x}_{k|k} = \sum_{j=1}^r \mu_k^j \hat{x}_{k|k}^j \quad (2.84)$$

$$P_{k|k} = \sum_{j=1}^r \mu_k^j \left[P_{k|k}^j + (\hat{x}_{k|k}^j - \hat{x}_{k|k})(\hat{x}_{k|k}^j - \hat{x}_{k|k})^T \right] \quad (2.85)$$

L'estimateur multi-modèles statique ou autonome suppose qu'un modèle parmi l'ensemble des modèles soit correct et qu'aucun changement de modèle n'ait eu lieu durant l'estimation. Par conséquent, la probabilité du modèle correct ou du modèle qui en est le plus proche converge vers l'unité [10]. Bien que l'estimateur AMM ait été employé avec succès dans beaucoup d'applications de poursuite de cibles [63], d'estimation de paramètres inconnus [86] et de détection

de défauts [11], son point faible reste son incapacité à s'adapter aux changements de modèles qui caractérisent les manœuvres d'une cible par exemple. De plus, le manque d'interactions entre les différents modèles fait que l'erreur dans les filtres associés aux modèles incorrects atteint des valeurs importantes qui risquent de faire diverger le filtre. Ces limitations sont traitées dans la seconde génération des estimateurs MM : les estimateurs MM dynamiques abordés dans la section suivante.

2.5 Les estimateurs multi-modèles dynamiques

Lorsque le système peut changer de modèle au cours du processus d'estimation, il est formalisé de la manière suivante :

$$x_k = F(M_k)x_{k-1} + v(k-1, M_k) \quad (2.86)$$

$$z_k = H(M_k)x_k + w(k, M_k) \quad (2.87)$$

Où M_k représente le modèle à l'instant k . Ce modèle appartient à l'ensemble des r modèles possibles :

$$M_k \in \mathbb{M} \quad \text{avec } \mathbb{M} = \{M^j\}_{j=1}^r \quad (2.88)$$

On suppose que le changement de modèle (ou le saut de modèle) est un processus de Markov (ou une chaîne de Markov) avec des probabilités de transition connues et invariables dans le temps :

$$p_{ij} \triangleq P(M_k = M^j | M_{k-1} = M^i) \quad (2.89)$$

A l'instant k , il existe r^k historiques possibles pour le système. Par exemple, si $r = 3$, à l'instant $k = 6$, il existe $3^6 = 729$ historiques possibles pour le système.

La méthode optimale pour calculer l'estimée à l'instant k doit prendre en considération tous les historiques possibles du système. Cependant, la croissance en puissance de k du nombre d'historiques est impossible à implémenter en pratique. La seule approche possible pour contourner cette augmentation exponentielle est le passage à des techniques sous-optimales qui reposent sur deux transformations :

1. *Elaguer* (*prune* en anglais) ou écarter les séquences de modèles invraisemblables et préserver les meilleures séquences. Ceci inclut la préservation de la meilleure séquence glo-

blement ou de la meilleure séquence pour chaque modèle. Les deux approches construisent un arbre d'historique tronqué et nécessitent un processus de décision dure.

2. *Fusionner* (*merge* en anglais) les séquences de modèles similaires et réinitialiser les différents filtres par des estimées qui englobent “approximativement” tout l'historique passé. On trouve deux variantes de cette approche qui utilise la décision douce :
 - L'algorithme pseudo-bayésien généralisé : GPB (*Generalized Pseudo-Bayesian*)
 - L'algorithme multi-modèles interactif : IMM (*Interacting Multiple-Model*)

L'idée générale de ces deux solutions est de remplacer l'arbre d'historique grandissant exponentiellement par un arbre simplifié qui préserve la plupart de l'information utile. Ces arbres sont illustrés pour les différents estimateurs MM étudiés par les figures 2.7, 2.9 et 2.11.

2.5.1 Estimateur Pseudo-bayésien généralisé du premier ordre (GPB1)

Dans l'estimateur GPB1, l'estimation à l'instant k est calculée à partir des r différents modèles possibles. Tous les historiques passés sont combinés.

On dénote par M_k^j l'événement que le modèle j existe en effet à l'instant k :

$$M_k^j \triangleq \{M_k = M^j\} \quad (2.90)$$

L'estimée est calculée en appliquant le théorème des probabilités totales :

$$\begin{aligned} p(x_k | Z_k) &= \sum_{j=1}^r p(x_k | M_k^j, Z_k) P(M_k^j | Z_k) \\ &= \sum_{j=1}^r p(x_k | M_k^j, z_k, Z_{k-1}) \mu_k^j \end{aligned} \quad (2.91)$$

L'hypothèse faite dans le cas de l'estimateur GPB1 consiste à supposer que tout le passé Z_{k-1} est résumé dans l'estimée précédente $\hat{x}_{k-1|k-1}$ et sa covariance $P_{k-1|k-1}$:

$$p(x_k | Z_k) \approx \sum_{j=1}^r p(x_k | M_k^j, z_k, \hat{x}_{k-1|k-1}, P_{k-1|k-1}) \mu_k^j \quad (2.92)$$

L'algorithme de l'estimation GPB1 est décrit par la figure 2.6. On illustre sur la figure 2.7 le cycle de fonctionnement de l'estimateur.

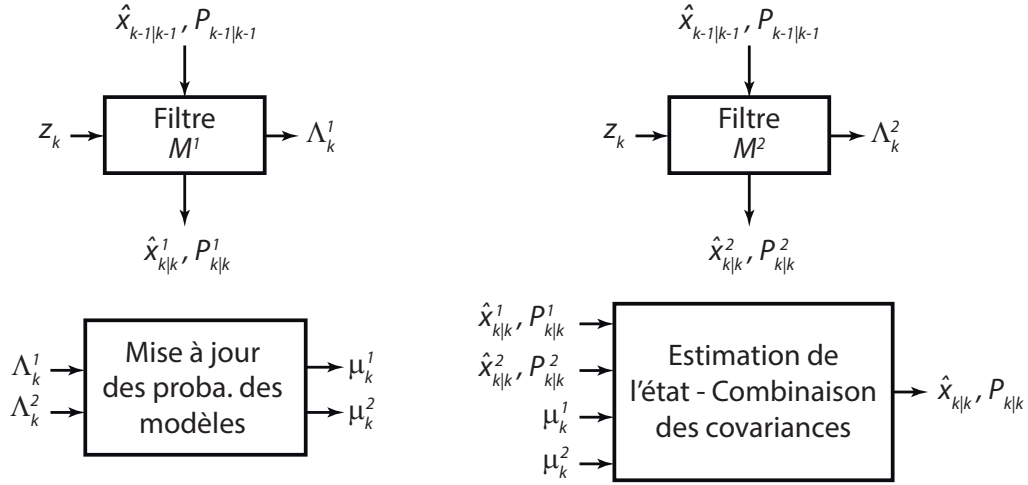


FIGURE 2.6 – Structure de l'estimateur GPB1 selon [9]

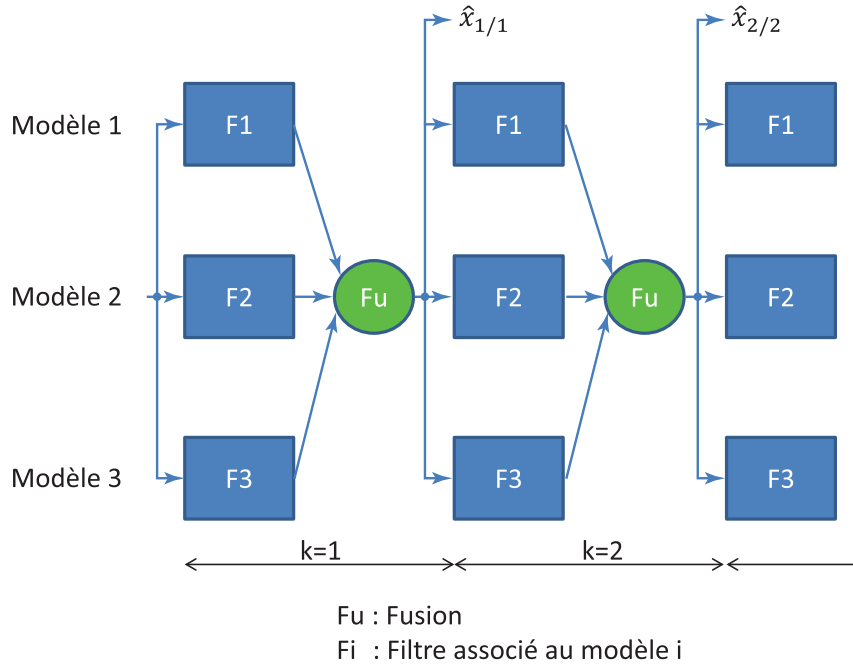


FIGURE 2.7 – Cycle de l'estimateur GPB1

2.5.2 Filtre Pseudo-bayésien généralisé du second ordre (GPB2)

L'estimateur GPB2 considère tout l'historique à l'instant $k-1$. Ainsi, l'estimation à l'instant k est calculée par tous les modèles possibles à l'instant actuel k et à l'instant précédent $k-1$, ce qui suppose le traitement de r^2 hypothèses différentes. Toutes les sorties des filtres sont fusionnées grâce à l'introduction d'une probabilité de transition entre modèles de l'état $k-1$ à k . Cette probabilité notée $\mu_{k-1|k}^{i|j}$ est définie par :

$$\mu_{k-1|k}^{i|j} \triangleq P(M_{k-1}^i | M_k^j, Z_k) \quad (2.93)$$

Elle indique la probabilité d'avoir le modèle i à l'instant $k - 1$ sachant que le modèle j est choisi à l'instant k .

L'algorithme de l'estimateur GPB2 est décrit par la figure 2.8. Sur cette figure $\hat{x}_{k|k}^{ij}$ représente l'estimée par le filtre M_j obtenue à partir de l'état $\hat{x}_{k-1|k-1}^i$. Le cycle d'évolution de l'estimateur GPB2 est illustré par la figure 2.9. Il nécessite r^2 filtres en parallèles (ici on prend $r = 3$).

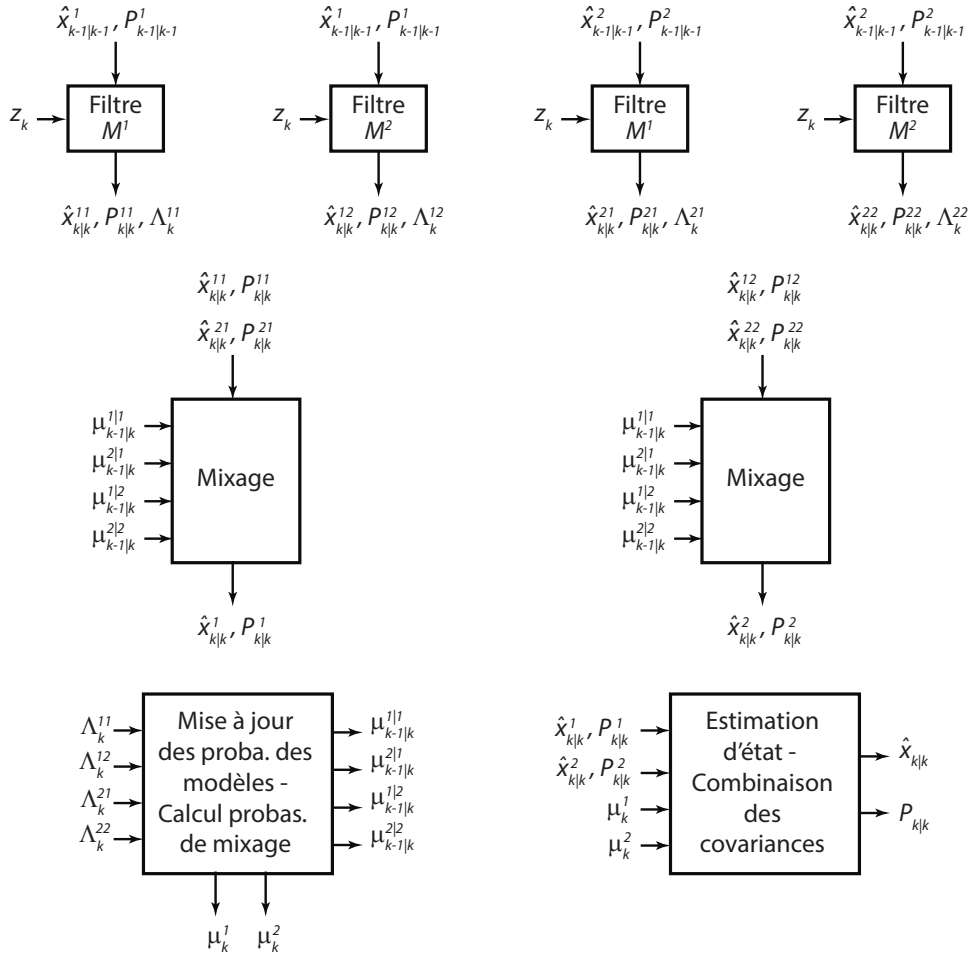


FIGURE 2.8 – Structure de l'estimateur GPB2 selon [9]

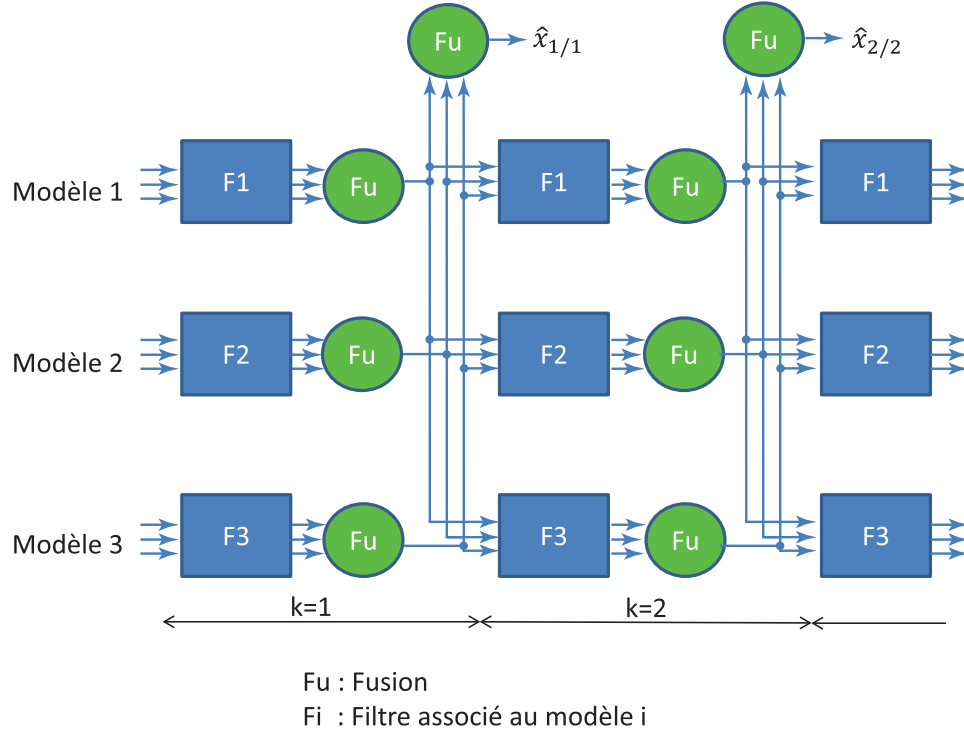


FIGURE 2.9 – Cycle de l'estimateur GPB2

2.5.3 Filtre multi-modèles interactif (IMM)

L'estimateur multi-modèles interactif (IMM : *Interacting Multiple Model*) réalise un bon compromis en réduisant la complexité de calcul trouvée dans le GPB2 qui considère un niveau supplémentaire de l'historique par rapport au GPB1. Il améliore l'initialisation des filtres en utilisant une interaction ou un mixage des estimées des différents filtres à l'instant précédent au lieu d'une valeur unique.

En employant le théorème de probabilité totale pour les r filtres évoluant en parallèle, on a :

$$\begin{aligned}
 p(x_k | Z_k) &= \sum_{j=1}^r p(x_k | M_k^j, Z_k) P(M_k^j | Z_k) \\
 &= \sum_{j=1}^r p(x_k | M_k^j, z_k, Z_{k-1}) \mu_k^j
 \end{aligned} \tag{2.94}$$

Le premier terme de cette somme est la loi de probabilité *a posteriori* de l'état conditionnée

par le modèle M_k^j . Elle peut être réécrite en utilisant le théorème de Bayes :

$$p(x_k | M_k^j, z_k, Z_{k-1}) = \frac{p(z_k | M_k^j, x_k)}{p(z_k | M_k^j, Z_{k-1})} p(x_k | M_k^j, Z_{k-1}) \quad (2.95)$$

L'équation (2.95) représente un cycle d'estimation du filtre de Kalman associé au modèle M_k^j en partant de la loi *a priori* qui est le terme complètement à droite de cette équation.

On applique maintenant le théorème de probabilité totale sur cette loi *a priori* :

$$p(x_k | M_k^j, Z_{k-1}) = \sum_{i=1}^r p(x_k | M_k^j, M_{k-1}^i, Z_{k-1}) P(M_{k-1}^i | M_k^j, Z_{k-1}) \quad (2.96)$$

L'approximation que l'on fait dans l'estimateur IMM consiste à considérer que l'historique Z_{k-1} est résumé dans les estimées fournies par les r modèles à l'instant $k-1$ et leur covariance :

$$\begin{aligned} p(x_k | M_k^j, Z_{k-1}) &\approx \sum_{i=1}^r p(x_k | M_k^j, M_{k-1}^i, \{\hat{x}_{k-1|k-1}^l, P_{k-1|k-1}^l\}_{l=1}^r) \mu_{k-1|k-1}^{i|j} \\ &= \sum_{i=1}^r p(x_k | M_k^j, M_{k-1}^i, \hat{x}_{k-1|k-1}^i, P_{k-1|k-1}^i) \mu_{k-1|k-1}^{i|j} \end{aligned} \quad (2.97)$$

Ainsi, la loi *a priori* de chaque filtre est un *mélange* pondéré avec les facteurs $\mu_{k-1|k-1}^{i|j}$ appelés *probabilités de mélange* ou de *mixage* :

$$\mu_{k-1|k-1}^{i|j} \triangleq P(M_{k-1}^i | M_k^j, Z_{k-1}) \quad (2.98)$$

Le mélange défini par (2.97) est supposé gaussien (somme de lois normales) et peut être approximé par une seule loi normale en utilisant les moments d'ordre 1 et 2 :

$$\begin{aligned} p(x_k | M_k^j, Z_{k-1}) &= \sum_{i=1}^r \mathcal{N}(x_k | E(x_k | M_k^j, \hat{x}_{k-1|k-1}^i), \text{cov}(\cdot)) \mu_{k-1|k-1}^{i|j} \\ &\approx \mathcal{N}(x_k | \sum_{i=1}^r E(x_k | M_k^j, \hat{x}_{k-1|k-1}^i) \mu_{k-1|k-1}^{i|j}, \text{cov}(\cdot)) \\ &= \mathcal{N}(x_k | E(x_k | M_k^j, \sum_{i=1}^r \hat{x}_{k-1|k-1}^i \mu_{k-1|k-1}^{i|j}), \text{cov}(\cdot)) \end{aligned} \quad (2.99)$$

Où $\text{cov}(\cdot)$ est le terme de covariance de la loi normale, non développé ici pour des raisons de simplification.

La dernière ligne de l'équation (2.99) découle de la linéarité du filtre de Kalman et nous amène au principe de l'estimateur IMM : L'entrée (ou l'initialisation) du filtre associée au modèle j est obtenue par une interaction entre les r filtres, ce qui consiste à *mélanger* ou *mixer* les estimées $\hat{x}_{k-1|k-1}^i$ avec les pondérations $\mu_{k-1|k-1}^{i|j}$.

La figure 2.10 évoque la structure de l'estimateur IMM, qui ressemble au GPB1 à la différence de l'ajout d'une étape préalable de mixage pour réinitialiser les filtres. Il est donc moins complexe d'utiliser r filtres interactifs, que d'utiliser le GPB2 qui exécute r^2 filtres.

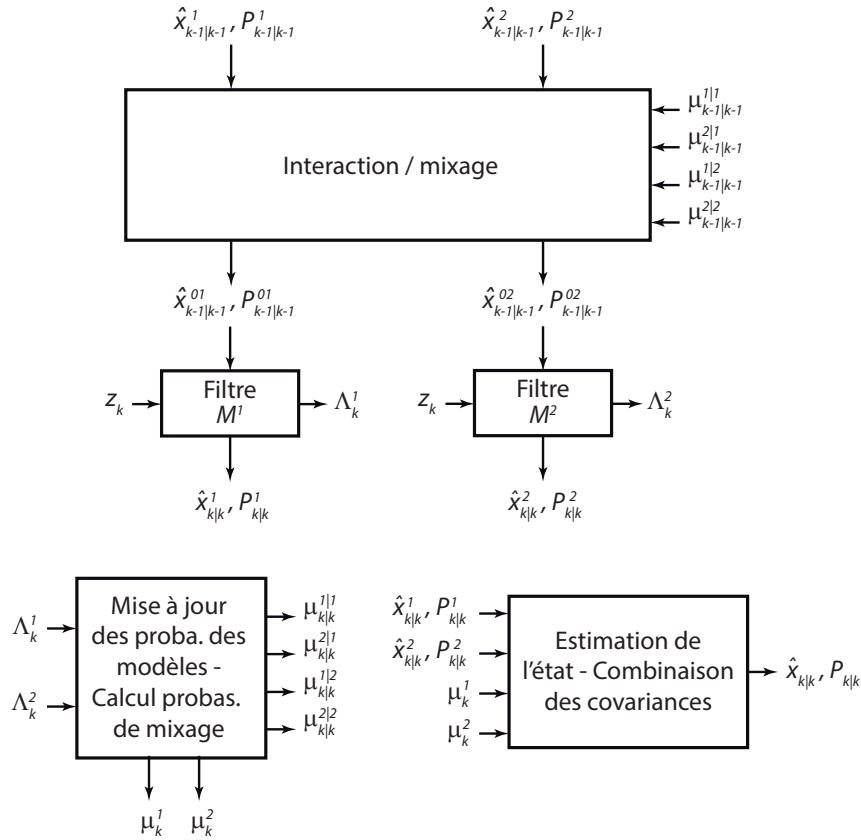


FIGURE 2.10 – Structure de l'estimateur IMM selon [9]

On illustre sur la figure 2.11 le cycle d'estimation du filtre IMM.

En comparant l'initialisation des filtres dans le GPB1, GPB2 et l'IMM, on constate ce qui suit : L'algorithme du GPB1 réinitialise chaque filtre avec l'unique "meilleure" statistique du passé qui est l'estimée globale à l'instant précédent (voir figure 2.7). L'interaction entre les filtres se fait uniquement à travers cette entrée commune à chaque itération. L'algorithme IMM,

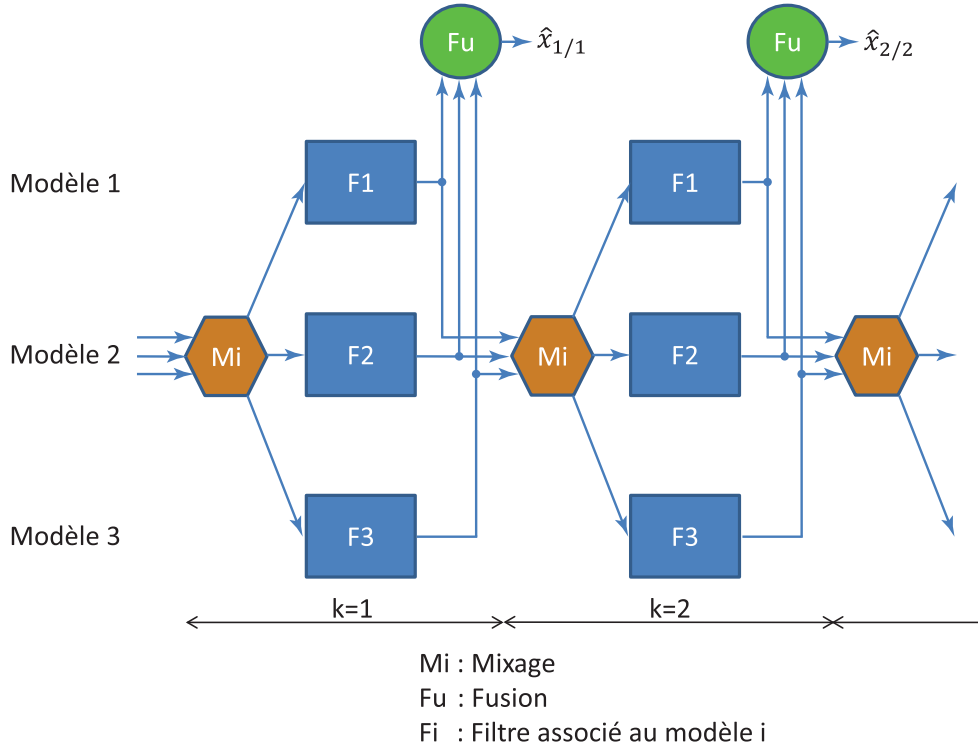


FIGURE 2.11 – Cycle de l'estimateur IMM

en revanche, fournit pour chaque filtre une réinitialisation individualisée, ce qui constitue la “meilleure” statistique de tout le passé sous l’hypothèse que le modèle M^j associé au filtre j soit actif à l’instant k (voir figure 2.11).

L’algorithme IMM applique le mixage avant le filtrage conditionnel, tandis que l’algorithme du GPB2 l’applique après le filtrage conditionnel. Ce qui résulte en une division par r du nombre d’hypothèses à considérer pour l’IMM comparé au GPB2.

L’algorithme de l’estimateur IMM est implémenté comme suit :

1. **Calcul des probabilités de mixage** : C’est la probabilité que le modèle M^i , conditionnée par les mesures Z_{k-1} , ait été choisi à l’instant $k-1$ sachant que le modèle M^j est sélectionné à l’instant k , :

$$\begin{aligned} \mu_{k-1|k-1}^{i,j} &\triangleq P(M_{k-1}^i | M_k^j, Z_{k-1}) \quad \text{avec } i, j = 1, \dots, r \\ &= \frac{1}{c_j} P(M_k^j | M_{k-1}^i, Z_{k-1}) P(M_{k-1}^i | Z_{k-1}) \end{aligned} \quad (2.100)$$

qui s'écrit aussi :

$$\mu_{k-1|k-1}^{i|j} = \frac{1}{\bar{c}_j} p_{ij} \mu_{k-1}^i \quad (2.101)$$

\bar{c}_j étant une constante de normalisation définie par :

$$\bar{c}_j = \sum_{i=1}^r p_{ij} \mu_{k-1}^i \quad (2.102)$$

Notons ici que les probabilités sont conditionnées par Z_{k-1} , il faudrait donc faire le mixage au début du cycle d'estimation et non pas à sa fin.

2. **Mixage** : En utilisant les estimées de chaque filtre à l'instant $k-1$ pondérées par les probabilités de mixage, on calcule l'état initial de chaque filtre en utilisant l'équation (2.99) :

$$\hat{x}_{k-1|k-1}^{0j} = \sum_{i=1}^r \hat{x}_{k-1|k-1}^i \mu_{k-1|k-1}^{i|j} \quad \text{avec } j = 1, \dots, r \quad (2.103)$$

La covariance correspondante est :

$$P_{k-1|k-1}^{0j} = \sum_{i=1}^r \mu_{k-1|k-1}^{i|j} \left[P_{k-1|k-1}^i + (\hat{x}_{k-1|k-1}^i - \hat{x}_{k-1|k-1}^{0j})(\hat{x}_{k-1|k-1}^i - \hat{x}_{k-1|k-1}^{0j})^T \right] \quad (2.104)$$

3. **Filtrage** : L'état initial de chaque filtre (2.103) et sa covariance (2.104) sont utilisés comme des entrées pour les filtres associés aux modèles M_k^j . Ces derniers utilisent les mesures z_k pour calculer les estimées $\hat{x}_{k|k}^j$ et leur covariance $P_{k|k}^j$. La fonction de vraisemblance pour chaque filtre s'écrit :

$$\Lambda_k^j = p(z_k | M_k^j, Z_{k-1}) \quad (2.105)$$

qui devient en utilisant l'approximation faite dans l'IMM :

$$\Lambda_k^j = p(z_k | M_k^j, \hat{x}_{k-1|k-1}^{0j}, P_{k-1|k-1}^{0j}) \quad (2.106)$$

qui s'écrit encore sachant que les filtres fonctionnent sous l'hypothèse gaussienne (voir

l'équation (2.81) pour plus de détails) :

$$\Lambda_k^j = \mathcal{N}(z_k | \hat{z}_{k|k-1}^j, S_k^j) \quad (2.107)$$

4. **Mise à jour des probabilités des modèles** : Ces probabilités sont calculées comme suit :

$$\begin{aligned} \mu_k^j &\triangleq P(M_k^j | Z_k) \quad \text{avec } j = 1, \dots, r \\ &= \frac{1}{c} p(z_k | M_k^j, Z_{k-1}) P(M_k^j | Z_{k-1}) \\ &= \frac{1}{c} \Lambda_k^j \sum_{i=1}^r P(M_k^j | M_{k-1}^i, Z_{k-1}) P(M_{k-1}^i | Z_{k-1}) \\ &= \frac{1}{c} \Lambda_k^j \sum_{i=1}^r p_{ij} \mu_{k-1}^i \end{aligned} \quad (2.108)$$

Ou encore d'après l'équation (2.102) :

$$\mu_k^j = \frac{1}{c} \Lambda_k^j \bar{c}_j \quad (2.109)$$

c étant une constante de normalisation définie par :

$$c = \sum_{j=1}^r \Lambda_k^j \bar{c}_j \quad (2.110)$$

5. **Combinaison des estimées** : L'estimée finale est une combinaison des estimées fournies par chaque modèle pondérée par la probabilité de chaque modèle :

$$\hat{x}_{k|k} = \sum_{j=1}^r \hat{x}_{k|k}^j \mu_k^j \quad (2.111)$$

$$P_{k|k} = \sum_{j=1}^r \mu_k^j \left[P_{k|k}^j + (\hat{x}_{k|k}^j - \hat{x}_{k|k})(\hat{x}_{k|k}^j - \hat{x}_{k|k})^T \right] \quad (2.112)$$

L'étape de combinaison sert à fournir l'estimée finale et ne fait pas partie de l'algorithme récursif de l'estimateur IMM.

Une variante de l'estimateur IMM est l'estimateur RIMM (*Reweighted IMM*) qui est décrit dans [68]. La structure de l'estimateur RIMM ressemble à la structure du filtre IMM, la différence réside essentiellement dans la formule de mixage. Les entrées sont mixées après l'étape de prédiction des filtres associés aux différents modèles et les r^2 transitions possibles.

2.6 Estimateur multi-modèles à structure variable (VSMM)

Tous les estimateurs multi-modèles abordés dans les sections précédentes se classent sous la catégorie des estimateurs dits à structure fixe ou FSMM (*Fixed Structure Multiple Model*). Cette appellation vient du fait que les modèles utilisés forment un seul ensemble invariable dans le temps. Durant les trois dernières décennies, les estimateurs FSMM ont fournis de très bons résultats dans la résolution des problèmes d'estimation hybride [54]. Inspirés par le succès de l'estimateur IMM, les chercheurs ont concentré leurs efforts sur plusieurs aspects des FSMM [47] : 1) développer des estimateurs FSMM implémentables en temps réel, 2) concevoir de meilleurs modèles pour des problèmes particuliers et 3) appliquer les techniques d'estimation MM dans la résolution de problèmes réels.

Même s'il y a toujours des possibilités de développement des FSMM, on ne peut s'attendre à des développements majeurs vu que ces estimateurs ont atteint leurs limites. Cette perception découle de la compréhension des différentes limitations des estimateurs FSMM dont on cite notamment [47] :

- Ils supposent que le système peut être représenté avec une précision suffisante par l'un des modèles appartenant à un ensemble de modèles déterminé en avance
- L'utilisation de plus de modèles que nécessaire détériore les performances
- Les ressources nécessaires au calcul augmentent considérablement lorsque le nombre de modèles utilisés augmente.

Un estimateur MM qui n'utilise pas toujours le même ensemble de modèles s'appelle un *Estimateur Multi-modèle à structure variable* ou VSMM (*Variable Structure Multiple Model*).

Selon [43] la différence clé entre les estimateurs VSMM et FSMM optimaux est que le premier est une somme pondérée par les probabilités de tous les estimateurs issus de la séquence de l'ensemble des modes admissibles mutuellement exclusifs et exhaustifs tandis que le second est une somme de tous les estimateurs des séquences de modes possibles.

Un concept clé dans l'estimation VSMM est que les ensembles de modes sont dépendants de l'état [44]. Dit autrement, étant donné le mode actuel (et l'état de base actuel), l'ensemble des modes possibles à l'instant suivant est un sous ensemble de l'espace des modes déterminés par la loi de transition [53]. Considérons une application de poursuite d'une voiture avec trois modèles : mouvement en ligne droite ($M^1 = 1$), rotation (ou virage) à gauche ($M^2 = 2$) et rotation à droite ($M^3 = 3$). Initialement ($k = 1$), la voiture se déplace en ligne droite sur une route. Elle arrive à une intersection avec quatre routes à $k = 10$ où elle peut continuer en ligne

droite ou tourner à gauche ou à droite. A $k = 11$, si elle a fait un virage à gauche à $k = 10$, la voiture peut soit se déplacer en ligne droite ou continuer sa rotation à gauche (faisant une rotation en U). Ensuite, elle continue en ligne droite jusqu'à ce qu'elle arrive à un espace ouvert à $k = 20$ où le mode de mouvement peut être n'importe lequel. A partir de cette description, les ensembles de modes dépendants de l'état jusqu'à $k = 20$ sont :

$$\begin{aligned} \mathbb{S}_1^1 &= \dots = \mathbb{S}_9^1 = \{1\}, \quad \mathbb{S}_{10}^1 = \{1, 2, 3\}, \quad \mathbb{S}_{11}^1 = \{1\} \\ \mathbb{S}_{11}^2 &= \{1, 2\}, \quad \mathbb{S}_{11}^3 = \{1\}, \quad \dots, \quad \mathbb{S}_{20}^i = \{1, 2, 3\} \quad \text{avec } i = 1, 2, 3 \end{aligned} \quad (2.113)$$

La séquence des ensembles de modes possibles jusqu'à $k = 20$ est :

$$\begin{aligned} \mathbb{S}^{20} &= \{\mathbb{S}_1, \dots, \mathbb{S}_{10}, \mathbb{S}_{11}, \mathbb{S}_{12}, \dots, \mathbb{S}_{20}\} \\ &= \{\{1\}, \dots, \{1, 2, 3\}, \{1, 2\}, \{1\}, \dots, \{1, 2, 3\}\} \end{aligned} \quad (2.114)$$

Où $\mathbb{S}_k = \cup_i \mathbb{S}_k^i$ est l'union des ensembles de modes dépendant de l'état à l'instant k . Notons que l'ensemble des modes possibles à l'instant k dépend du mode s_{k-1} à l'instant $k - 1$ et des états de base x_{k-1} et x_k . Il a été démontré dans [44] que cet estimateur MM ne peut pas être optimal si à un instant k il utilise un ensemble de modèles \mathbb{M}_k différent de l'espace de modes \mathbb{S}_k . De ce fait, l'utilisation d'un ensemble fixe de modèles, soit $\mathbb{M} = \{1, 2, 3\}$, n'est pas préférable.

Une formulation reposant sur la théorie des graphes a été proposée dans [44], où le mode et la possibilité de transition d'un mode à un autre a été représentée respectivement par un nœud et un lien orienté, résultant en un graphe orienté qui indique l'association entre l'ensemble de modes et leur dépendance de l'état. Cette formulation a plusieurs avantages selon [44] et est la base d'une classe d'algorithmes VSMM.

2.6.1 L'approche d'adaptation récursive de l'ensemble des modèles (RAMS)

Dans la plupart des applications, on essaie de trouver la "meilleure" séquence des ensembles de modèles. L'approche RAMS (*Recursive Adaptive Model-Set*) [46, 44] a pour objectif de définir à chaque instant le meilleur ensemble de modèle. Cette détermination est réalisée récursivement. Chaque itération de l'algorithme RAMS consiste en deux étapes :

1. Le choix de l'ensemble de modèles (*Model-set adaptation*) : détermine à chaque instant l'ensemble de modèles à utiliser dans l'estimation MM en s'appuyant sur l'information *a posteriori* et la connaissance *a priori*. Les algorithmes RAMS se différencient par la manière avec laquelle l'ensemble des modèles est choisi.

2. L'estimation conditionnée par la séquence des ensembles des modèles : a pour objectif de trouver la meilleure estimée possible à partir d'une séquence d'ensemble de modèles. Elle consiste à 1) initialisation : affecter les probabilités initiales aux nouveaux modèles et initialiser les filtres qui s'y rapportent. Cette étape n'existe pas dans les deux premières générations d'estimateurs MM 2) estimation : utilisation de la stratégie de mixage, de fusion et de filtrage, similaire aux premières générations d'estimateurs MM.

L'adaptation des ensembles de modèles peut être décomposée en deux cas : l'expansion de l'ensemble de modèles ou sa réduction [46]. Cette décomposition possède d'importants avantages par rapport à la technique de commutation de l'ensemble de modèles en termes de performance [56, 46, 47]. L'expansion de l'ensemble de modèles est souvent plus importante que sa réduction. Cela est dû au fait que même si l'inclusion d'un modèle impossible peut être aussi mauvaise que l'exclusion d'un modèle possible, les performances d'un algorithme MM souffrent considérablement si un modèle fortement probable est écarté, tandis que les performances ne se dégradent que de peu si un modèle fortement improbable est inclus. Malheureusement, l'expansion de l'ensemble de modèles est en général plus difficile que sa réduction [53].

L'expansion et la réduction d'un ensemble de modèles nécessite deux tâches fonctionnelles : la “*candidature*” qui détermine les ensembles candidats pour l'expansion ou la réduction, et la “*décision*” qui retient le(s) meilleur(s) ensemble(s) candidat(s). La phase de candidature pour l'expansion entraîne l'activation ou la génération d'un ensemble de nouveaux modèles, ce qui est la tâche principale de tout algorithme d'adaptation des ensembles de modèles. Cette phase de candidature est plus facile dans la réduction que dans l'expansion.

2.6.2 Décision sur un ensemble de modèles candidats

La décision sur un ensemble de modèles candidats peut être formulée comme un problème statistique de décision, plus particulièrement, un problème de test statistique d'hypothèses dans un cadre séquentiel, ce qui est naturel puisque les observations sont disponibles séquentiellement. Étant donné que les hypothèses sont généralement fixes pour le test d'hypothèses, le vrai mode noté s , ainsi que l'ensemble des modèles, sont supposés constants pendant la période où le test est effectué.

Probabilité et vraisemblance d'un ensemble de modèles

Étant donné que la tâche la plus importante dans un estimateur VSMM est de décider quel est le bon ensemble de modèles, les probabilités et/ou les vraisemblances des ensembles des modèles en question sont d'une importance majeure. La vraisemblance marginale d'un ensemble

de modèles \mathbb{M} à l'instant k est la somme des probabilités prédites $P(M_k^i | s \in \mathbb{M}, Z_{k-1})$ multipliée par les vraisemblances marginales $f(\tilde{z}_k | s = M^i, Z_{k-1})$ de tous les modèles M^i dans \mathbb{M} [46, 47, 56] :

$$\begin{aligned} L_k^{\mathbb{M}} &\triangleq f(\tilde{z}_k | s \in \mathbb{M}, Z_{k-1}) \\ &= \sum_{M^i \in \mathbb{M}} f(\tilde{z}_k | s = M^i, Z_{k-1}) P(s = M^i | s \in \mathbb{M}, Z_{k-1}) \end{aligned} \quad (2.115)$$

Où \tilde{z}_k est le résidu des mesures. La vraisemblance jointe de l'ensemble de modèles \mathbb{M} jusqu'à l'instant k est définie par $L_{\mathbb{M}}^k \triangleq f(\tilde{z}_k | s \in \mathbb{M})$. Soit Λ^k le ratio de la vraisemblance jointe de l'ensemble de modèles \mathbb{M}_1 à \mathbb{M}_2 qui est égal au produit des ratios des vraisemblances marginales des ensembles de modèles [46] :

$$\Lambda^k = \frac{L_{\mathbb{M}_1}^k}{L_{\mathbb{M}_2}^k} = \prod_{k_0 \leq \kappa \leq k} \frac{L_{\kappa}^{\mathbb{M}_1}}{L_{\kappa}^{\mathbb{M}_2}} \quad (2.116)$$

Où k_0 est l'instant de début du test. La probabilité postérieure indiquant que le vrai mode est inclus dans \mathbb{M} est définie par :

$$\begin{aligned} \mu_k^{\mathbb{M}} &\triangleq P(s \in \mathbb{M} | s \in \mathcal{M}, Z_k) \\ &= \sum_{M^i \in \mathbb{M}} P(M_k^i | s \in \mathcal{M}, Z_k) = \sum_{M^i \in \mathbb{M}} \mu_k^i \end{aligned} \quad (2.117)$$

C'est la somme des probabilités de tous les modes dans \mathbb{M} , où \mathcal{M} est l'union de tous les ensembles de modèles considérés y compris \mathbb{M} comme un sous-ensemble. La probabilité du mode μ_k^i provient de l'estimateur MM utilisant \mathcal{M} .

Plusieurs tests d'hypothèses ont été proposés dans [46], [47] et appliqués dans [56], [47], [55]. ils s'appuient sur les vraisemblances ou les probabilités des ensembles de modèles. D'autres sujets relatifs aux estimateurs VSMM sont discutés dans les références [50], [52], [51], [53], [47], [46], [44] comme notamment l'initialisation des nouveaux modèles récemment sélectionnés, les différents algorithmes VSMM, les méthodes de prise de décision ainsi que des applications des VSMM.

2.7 Conclusion

Dans ce chapitre nous avons fait un état de l'art de l'estimation Multi-Modèle. Cette technique est principalement appliquée à la poursuite (*tracking*) des cibles manœuvrantes. Les méthodes non-MM sont optimales si le système peut être représenté par un modèle unique couvrant toute sa dynamique. Dans un bon nombre d'applications, ce modèle unique n'est pas possible, soit parce que ce modèle unique est trop complexe, soit parce que le système peut opérer avec plusieurs modes de fonctionnement différents pouvant être représentés par des modèles plus simples. C'est là que les estimateurs MM prouvent leur efficacité. La première génération d'estimateurs autonomes ou statiques AMM a une structure composée de plusieurs filtres élémentaires indépendants. Son avantage par rapport aux approches non-MM réside dans sa capacité à traiter les résultats de plusieurs filtres en parallèle et de définir la probabilité *a posteriori* du modèle sachant les mesures qui permet de choisir le modèle le plus adapté au système observé. Cette première génération est peu performante aux instants de transitions entre modèles à cause du manque d'interaction entre les filtres.

La deuxième génération des estimateurs MM dynamiques ou coopératifs exploite des stratégies efficaces de coopération entre les filtres tout en héritant des avantages de la première génération liée au calcul de la sortie. Elle est représentée principalement par le filtre IMM. Le filtre IMM a connu un grand succès car il réalise un bon compromis entre complexité et efficacité. Il a été appliqué dans la résolution d'un bon nombre de problèmes de poursuite de cibles manœuvrantes. Cependant, l'une des limitations de cette approche est l'utilisation d'un ensemble fixe de modèles et de supposer que le vrai modèle appartient à l'ensemble de modèles considérés. Cette génération n'est pas adaptée aux environnements où aucun des filtres élémentaires ne correspond au mode actuel du système ou lorsque le trop grand nombre de modèles crée une concurrence excessive qui dégrade les performances de l'estimateur.

La troisième génération VSMM s'appuie sur une architecture plus ouverte que ses prédécesseurs. Elle hérite de la seconde génération une stratégie efficace de coopération entre les filtres et elle hérite de la première génération des avantages liés à la méthode d'obtention de l'estimée finale. Elle s'adapte à l'environnement extérieur, d'une part par l'activation de nouveaux filtres élémentaires si les filtres existants ne sont pas assez bons, et d'autre part par la désactivation des filtres non adaptés à l'état actuel du système qui risquent de devenir une menace pour la qualité de l'estimation. Les recherches en cours pour cette troisième génération concernent les règles de décision pour l'activation et la désactivation des modèles qui sont associés aux filtres.

L'étude comparative réalisée dans [68] a montré qu'il n'existe pas d'algorithme optimal, et que les performances dépendent de l'application. En général, les algorithmes IMM, GPB2, VMM et BMM ont dépassé en performances les algorithmes AMM, GPB1 et RIMM. Les algorithmes de décision dure BMM et VMM comparés à ceux reposant sur la décision douce (GPB2, IMM) ont montré une légère amélioration de performance dans les cas où le vrai modèle de mouvement de la cible existait parmi l'ensemble des modèles. Cependant, ils étaient moins performants lorsque le vrai modèle de mouvement n'existait pas explicitement dans l'ensemble des modèles. Ce dernier cas est le plus important car souvent les vrais modèles de mouvement de la cible ne sont pas connus avec précision.

Enfin, cette étude a démontré que l'algorithme IMM possède le meilleur coût de calcul parmi les algorithmes qui présentaient des erreurs de poursuite acceptables et qu'il a montré une robustesse remarquable en cas d'inadéquation entre les modèles.

Chapitre 3

Map matching particulière

3.1 Introduction

La qualité du Map matching est limitée par plusieurs restrictions abordées au chapitre 1. L'une des limitations principales est le masquage des signaux des satellites GPS qui peut rendre le calcul de la position absolue impossible. C'est un phénomène courant dans les milieux urbains où les signaux satellitaires peuvent être masqués par de hauts bâtiments, ou lorsque le véhicule se trouve dans un garage ou un tunnel par exemple. L'utilisation de capteurs proprioceptifs tels que les odomètres, gyroscopes, accéléromètres, ... peut aider temporairement à localiser le véhicule durant l'absence des signaux GPS. Ce processus de localisation est communément connu sous le nom de *Localisation par l'estime* (DR : *Dead Reckoning* en anglais) où la position actuelle est calculée sur la base de la position précédente. Cependant, le niveau de bruit sur les mesures issues de ces capteurs réduit la qualité de ce type de localisation et résulte en une dérive de la position calculée avec le temps suite à l'accumulation des erreurs à chaque pas de calcul.

La mesure de la direction du véhicule améliore d'une façon significative l'association de la position actuelle du véhicule à un segment sur le réseau des routes de la carte numérique. Ceci a été démontré par Quddus dans l'algorithme topologique proposé dans [70] et qui repose sur le calcul d'une somme de pondérations pour chaque segment dépendant de plusieurs critères : la différence entre la direction du véhicule et la direction du segment, la proximité de la position du véhicule au segment et la position angulaire relative du segment. Il a conclu que parmi tous ces facteurs, la mesure de la direction du véhicule avait le plus grand impact sur la qualité du map matching. Cependant, il a admis que la précision de la mesure de direction issue d'un récepteur GPS se dégrade lorsque la vitesse du véhicule est faible et peut devenir indisponible

en cas de masquage de satellites GPS.

Afin de pallier cet inconvénient, la mesure de la direction peut être assurée par un magnétomètre qui offre une fréquence de mesure plus élevée et une autonomie vis-à-vis de la disponibilité des signaux GPS. Il a été démontré dans [78] et [26] que l'estimation sur la mesure de la direction peut être améliorée par l'implémentation de filtres circulaires se rapportant à la distribution de probabilité de von Mises. Ces filtres sont plus adaptés aux mesures angulaires et sont plus robustes aux transitions entre $-\pi$ et π qui se produisent toujours dans les mesures d'angles.

Dans ce chapitre, nous proposons une méthode novatrice de map matching qui met en correspondance les informations de direction présentes sur la carte et mesurées par un magnétomètre. Cette méthode repose sur un filtre particulière utilisant les mesures d'angle et de vitesse provenant de capteurs proprioceptifs montés sur le véhicule. Cette méthode permet d'assurer la continuité du processus de Map matching en cas d'absence des signaux GPS.

Nous allons suivre l'organisation suivante dans ce chapitre : après une description de la problématique actuelle de Map matching en absence des signaux GPS, nous présentons l'implémentation du filtre particulière pour l'estimation de la position du véhicule sur le segment. Nous évaluerons ensuite les performances de ce filtre en le comparant à l'algorithme de Map matching topologique proposé dans [70].

3.2 Position du problème

3.2.1 Les limites des approches classiques

Nous avons exposé dans la section 1.4.2 du Chapitre 1 un algorithme proposé par Quddus [70] utilisant une approche topologique améliorée de map matching. Yang a démontré dans [89] que l'approche topologique fournit de meilleurs résultats par rapport à d'autres approches, comme l'approche géométrique, car elle prend en compte la topologie et la structure du réseau de routes et élimine automatiquement les cas aberrants, comme par exemple un matching qui saute brusquement entre deux routes parallèles si seule la proximité des segments aux points GPS est considérée (voir figure 1.4). D'autres cas d'erreurs de matching ont été détaillés dans la section 1.3 du chapitre 1. Ces erreurs sont souvent dues à la non prise en considération de la topologie des routes.

Cet algorithme topologique possède des limitations non négligeables dont on cite :

1. *Matching du premier point* : Même si l'algorithme est topologique, le matching de la première position du véhicule utilise une approche purement géométrique avec comme seul critère la proximité de la position mesurée aux nœuds du réseau de routes. Ceci est un point faible car un mauvais choix du nœud et par la suite du segment de départ implique un mauvais matching des points suivants car l'approche topologique dans cet algorithme préconise de continuer le matching sur le même segment jusqu'à détection de sa fin ou d'une manœuvre de rotation. Ceci est illustré sur la figure 3.1 où une erreur GPS sur le point P^1 provoque un mauvais choix du nœud de départ B et par la suite du segment BC au lieu de DE .

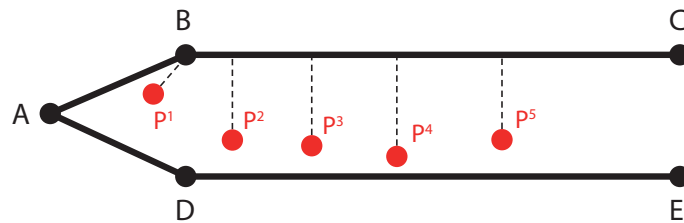


FIGURE 3.1 – Problème du matching sur une jonction en Y

2. *Mesure de la direction du véhicule* : Cette mesure repose uniquement sur les points GPS. La direction est l'angle entre le Nord et la ligne formée par le point GPS actuel et le précédent. Cependant à cause des différentes erreurs liées au système GPS dans les milieux urbains (mauvais HDOP, multi-trajet des signaux GPS, faible nombre de satellites visibles), cette mesure de direction est imprécise. De plus, si la période d'échantillonnage du récepteur GPS est grande, les points GPS deviennent trop espacés, et cette mesure de direction devient très imprécise comme dans le cas de la figure 3.2 qui illustre la mesure de la direction juste après un virage. Ces erreurs de mesure sur la direction peuvent aussi être à l'origine d'une fausse alerte de manœuvre du véhicule et entraîner le retour de l'algorithme à l'étape initiale de recherche du nœud le plus proche.
3. *Basse fréquence d'échantillonnage du récepteur GPS* : Des points GPS espacés par des intervalles de temps trop longs rendent difficile voire impossible la reconnaissance de la route réelle que le véhicule a emprunté.
4. *Masquage des signaux GPS* : C'est un phénomène courant dans les milieux urbains où le récepteur GPS peut perdre partiellement ou complètement les signaux des satellites GPS à cause d'un obstacle obstruant le ciel (haut bâtiment, pont, arbre, garage, tunnel, ...). Un autre problème remarquable dans les milieux urbains impose que, lorsque certains

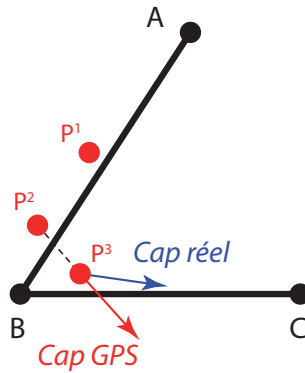


FIGURE 3.2 – Direction GPS comparée à la direction réelle

satellites sont masqués et que d'autres deviennent visibles, il est nécessaire de recourir à la ré-acquisition des satellites visibles. Ce traitement impose d'attendre une durée non négligeable avant de pouvoir utiliser les signaux des satellites pour le calcul de la position. Cela provoque une absence de localisation pouvant atteindre plusieurs minutes. Pendant ce temps, l'algorithme de Map matching est incapable d'estimer la position du véhicule.

5. *Utilisation de la décision dure* : Un autre aspect qui peut créer des défauts de matching concerne l'algorithme qui doit s'appuyer sur un processus de décision dure (*hard decision*) reposant sur un calcul de score. Le segment ayant le score le plus élevé est considéré comme étant le segment correct. Bien que cette approche soit suffisante dans beaucoup de cas, elle peut induire une fausse décision comme illustré sur la figure 3.1 où les segments *AB* et *AD* forment une jonction en Y et ont des scores très similaires au voisinage de la jonction. Si l'algorithme choisit par erreur le faux segment *AB*, il continuera sur ce même segment pendant un certain temps et ne cherchera à corriger le matching que lorsque l'écart entre la direction du véhicule et la direction du segment dépassera 45° . Il serait plus intéressant dans ce cas d'utiliser un processus de décision douce (*soft decision*) qui n'écarte pas les hypothèses potentielles mais les garde en vue et les adopter lorsque leur probabilité devient prépondérante. C'est l'une des forces de l'algorithme que nous proposons dans ce chapitre.

La limitation la plus sérieuse rencontrée est l'indisponibilité de la position GPS suite à un masquage des signaux satellitaires, ou pendant la ré-acquisition des signaux des nouveaux satellites. Dans ce cas, on a recours à la localisation par l'estime qui utilise les capteurs proprioceptifs installés sur le véhicule conjointement avec l'ancienne position estimée pour calculer la position actuelle.

3.2.2 Localisation par l'estime ou *Dead Reckoning*

Le principe de la localisation par l'estime est expliqué dans la figure 3.3. Soit $P(t)$ et $P(t+1)$ les positions du véhicule aux instants t et $t+1$ respectivement. Le véhicule étant équipé d'un capteur de vitesse fournissant la vitesse instantanée $v(t)$ et un capteur de direction instantanée $\theta(t)$, les coordonnées $x(t+1)$ et $y(t+1)$ sont calculées à partir des coordonnées à l'instant t : $x(t)$ et $y(t)$ par les équations (3.1) et (3.2) :

$$x(t+1) = x(t) + v(t) \Delta t \cos(\theta(t)) + dx(t) \quad (3.1)$$

$$y(t+1) = y(t) + v(t) \Delta t \sin(\theta(t)) + dy(t) \quad (3.2)$$

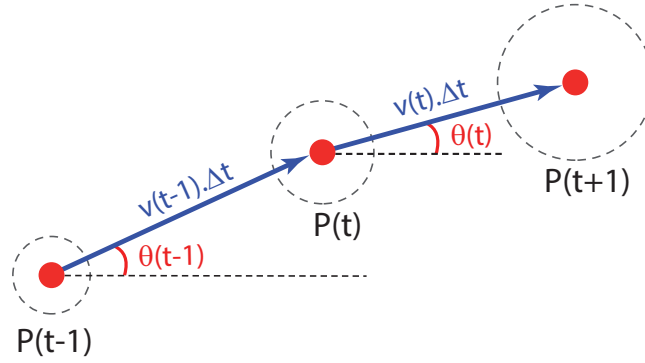


FIGURE 3.3 – Principe de la localisation par l'estime

Les termes $dx(t)$ et $dy(t)$ désignent l'erreur sur les composantes horizontale et verticale de la nouvelle position calculée. Il a été démontré dans [70] que ces erreurs dépendent de l'erreur sur l'angle $d\theta(t)$ et sur la vitesse $dv(t)$ tel que décrit par les équations (3.3) et (3.4) :

$$dx(t) = dv(t) \Delta t \cos(\theta(t)) - v(t) \Delta t \sin(\theta(t)) d\theta(t) \quad (3.3)$$

$$dy(t) = dv(t) \Delta t \sin(\theta(t)) + v(t) \Delta t d\theta(t) \cos(\theta(t)) \quad (3.4)$$

Il apparaît clairement que l'erreur (illustrée sur la figure 3.3 par un cercle en tirets) augmente à chaque pas de mesure. Cette erreur est d'autant plus importante que le biais du bruit sur la mesure de vitesse est grand. Ainsi, après une longue période, la position calculée devient imprécise ce qui diminue considérablement la qualité du Map matching qui repose sur le calcul de la position et la direction du véhicule.

3.2.3 L'approche proposée

Le filtre particulaire, connu aussi sous le nom de filtre de Monte-Carlo séquentiel, permet d'estimer la densité de probabilité *a posteriori* de l'état sachant les mesures. Il est défini dans un cadre bayésien et implémenté sous la forme d'équations récursives sans aucune restriction sur les dynamiques dans l'espace d'état, ni sur la densité de probabilité. Le filtre particulaire s'applique dans le cas d'équations d'état et de mesures linéaires ou non-linéaires. L'état initial ainsi que le bruit peuvent avoir des distributions quelconques. Le filtre particulaire utilise un ensemble de particules pour représenter la distribution *a posteriori* du vecteur d'état. Chaque particule possède un poids qui est associé à sa probabilité. Dans ce travail, le filtre particulaire est mis en œuvre pour trouver la position du véhicule sur la carte numérique lorsque les signaux GPS sont temporairement indisponibles. Ce filtre utilise dans ce cas les mesures de direction et de vitesse issues des capteurs montés sur le véhicule comme informations à mettre en correspondance avec la carte numérique.

Les premières particules sont délimitées par le cercle d'incertitude relatif à la dernière position GPS disponible comme illustré sur la figure 3.4. Ce cercle est une ellipse car les précisions horizontale et verticale du récepteur GPS ne sont pas identiques. Les segments choisis sont inscrits dans le cercle d'incertitude, et le segment retenu est celui qui répond le mieux aux critères définis dans [70].

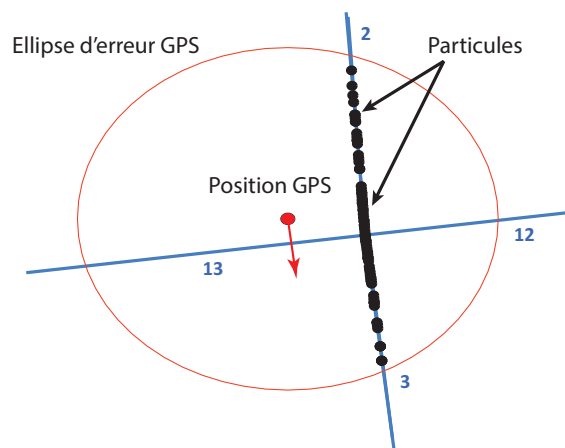


FIGURE 3.4 – Distribution initiale des particules

Les particules sont générées uniformément sur le segment retenu, puis un poids leur est affecté selon une distribution normale centrée sur la position du point GPS projeté sur le segment.

La variance de la distribution est celle de l'erreur GPS ramenée à la direction du segment.

Lors de l'étape de prédiction, chaque particule avance sur le segment sur lequel elle se trouve d'une distance : $d = v(t) \cdot \Delta t$, où $v(t)$ est la vitesse instantanée du véhicule à l'instant t et Δt est l'écart entre deux mesures du capteur de vitesse. Un bruit qui représente l'incertitude du modèle d'évolution sur le segment est ajouté à chaque déplacement de la particule.

Lorsque la particule atteint la fin du segment, elle est repositionnée sur l'un des segments connectés à la fin du segment courant. Le choix du segment est effectué aléatoirement suivant une distribution uniforme. La position de la particule sur le segment est obtenue par projection comme présenté figure 3.5.

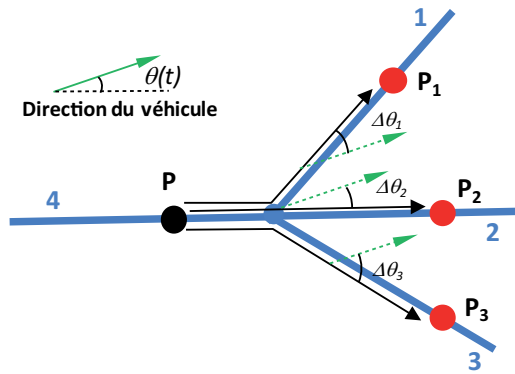


FIGURE 3.5 – Particule atteignant la fin d'un segment

L'étape de mise à jour du filtre particulaire permet de vérifier si la particule est sur le segment correct en lui assignant un poids qui dépend de la différence angulaire entre la direction du véhicule $\theta(t)$ et la direction α du segment sur lequel la particule se trouve à l'instant t . Ce poids donne plus d'importance aux particules sur des segments situés dans la direction du mouvement du véhicule et défavorise les particules évoluant sur des segments de directions différentes comme présenté figure 3.6. On utilise pour le calcul du poids la distribution de von Mises, qui est la loi normale dans le cas de variables angulaires.

Le segment comportant la particule de plus grand poids est le segment le plus probable. On ré-échantillonne les particules en donnant plus de probabilité à celles qui ont un poids plus important. Ainsi, on obtient un nouvel ensemble de particules plus concentré au voisinage de la position estimée comme illustré sur la figure 3.4.

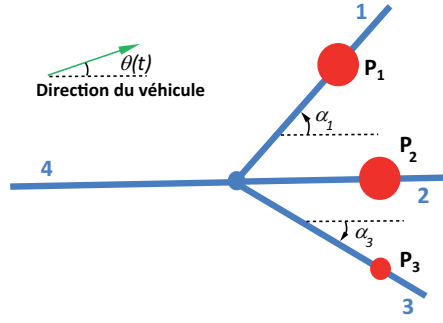


FIGURE 3.6 – Affectation des poids aux particules

On doit souligner ici que les parties clés de cet algorithme sont définies lorsque le véhicule effectue une manœuvre de rotation. Cette manœuvre aide l'algorithme à concentrer les particules autour de la position du changement de segment. En effet lorsque toutes les particules sont situées sur un même segment leurs poids est identique.

3.3 Map matching circulaire par filtrage particulaire SIR

3.3.1 Modélisation d'état

Dans un système de Map matching classique on peut utiliser un capteur GPS, un capteur de mesure de vitesse, et un magnétomètre pour la mesure de la direction du véhicule. On considère dans le cas du Map matching une position bidimensionnelle sur le plan de la carte.

Le modèle d'état, défini pour une localisation qui n'intègre pas les informations de la carte, est donné par les équations suivantes :

$$x(t) = x(t-1) + v(t) \Delta t \cos(\theta(t)) + \eta_x \quad (3.5)$$

$$y(t) = y(t-1) + v(t) \Delta t \sin(\theta(t)) + \eta_y \quad (3.6)$$

Dans ce système $v(t)$ la vitesse et $\theta(t)$ la direction du véhicule sont des commandes. η_x et η_y sont des bruits gaussien centrés. Δt est le pas temporel du système (durée entre deux mesures), et $v(t) \Delta t$ la distance parcourue. Il est montré dans [6] que $\theta(t)$ peut aussi être estimé avec un

filtre circulaire, on a alors le modèle d'état décrit par le système d'équations suivantes :

$$x(t) = x(t-1) + v(t) \Delta t \cos(\theta(t-1)) + \eta_x \quad (3.7)$$

$$y(t) = y(t-1) + v(t) \Delta t \sin(\theta(t-1)) + \eta_y \quad (3.8)$$

$$\tilde{\theta}(t) = \tilde{\theta}(t-1) + \eta_{\tilde{\theta}} \quad (3.9)$$

$$\theta(t) = \theta(t-1) + \tilde{\theta}(t-1) + \eta_{\theta} \quad (3.10)$$

avec $\tilde{\theta}(t)$ la variation d'angle pendant Δt . $\eta_{\tilde{\theta}}$ et η_{θ} sont des bruits normaux circulaires distribués suivant une loi de von Mises. On peut alors estimer les composantes du vecteur d'état qui sont les positions $x(t), y(t)$ et la direction $\theta(t), \tilde{\theta}(t)$ avec deux filtre indépendants : un filtre de Kalman étendu pour la position [35] et un filtre circulaire pour la direction. Dans ce cas les mesures utilisées sont la position GPS, et la direction du véhicule donnée par le magnétomètre.

Pour le Map matching, on considère une variable d'état supplémentaire $r(t)$ qui donne l'identité de la route courante, soit son numéro qui l'identifie dans la base de donnée cartographique. On peut à partir de ce numéro définir la direction $\alpha(r(t))$ et les coordonnées de début et fin du segment qui positionne la route sur la carte. On définit alors le modèle d'état décrit par les équations suivantes :

$$x(t) = x(t-1) + v(t) \Delta t \cos(\alpha(r(t-1))) + \eta_x \cos(\alpha(r(t-1))) \quad (3.11)$$

$$y(t) = y(t-1) + v(t) \Delta t \sin(\alpha(r(t-1))) + \eta_y \sin(\alpha(r(t-1))) \quad (3.12)$$

$$r(t) = r(t-1) + \eta_r \quad (3.13)$$

où η_r est un bruit discret uniforme sur les numéros de segments connectés au segment courant.

On suppose ici que la plate-forme mobile, équipée des capteurs, est positionnée sur la route et que les positions GPS sont indisponibles. On a alors l'équation suivante qui modélise les mesures fournies par le magnétomètre :

$$\phi(t) = \alpha(r(t)) + \eta_{\phi} \quad (3.14)$$

où η_{ϕ} est un bruit normal circulaire sur la mesure de direction.

On peut à partir des équations d'état (3.11), (3.12) et (3.13) et de l'équation de mesure (3.14) envisager l'implantation d'un filtre récursif sur le modèle d'un filtre de Kalman étendu. Dans

ce cas les deux étapes de fonctionnement du filtre sont les suivantes :

1. Prédiction : On prédit la position et l'identité du segment avec les équations d'états (3.11), (3.12) et (3.13).
2. Correction : On corrige l'identité du segment avec la mesure de direction et on corrige la position prédite en la projetant sur le segment identifié. En effet, la valeur de $r(t)$ estimée est celle qui minimise l'innovation du filtre (soit la différence entre $\phi(t)$ et $\alpha(r(t))$).

Le système est donc non linéaire par la modélisation d'état et par l'équation de mesure qui lie les observations aux variables d'états. Les fortes non-linéarités du système ne permettent pas de le réaliser sous la forme d'un filtre de Kalman étendu.

Nous proposons de réaliser l'estimateur de la loi *a posteriori* de l'état sachant les mesures, sous la forme d'un filtre particulière circulaire. Le choix du filtre particulière est motivé par la non linéarité du système. Le choix d'une description dans le domaine circulaire est motivé par l'utilisation d'observations angulaires comprises entre $-\pi$ et π . On présente dans le paragraphe suivant la distribution de von Mises du domaine circulaire utilisée pour modéliser le bruit sur les observations angulaires et pour définir le poids des particules du filtre.

3.3.2 Modélisation circulaire

On fait appel à la mesure d'angles dans beaucoup d'applications comme : la direction d'un véhicule, le sens du vent, la phase d'un signal, ... *etc.* Cette mesure est périodique, souvent exprimée en radians sur un intervalle défini : $[0, 2\pi[$ ou $[-\pi, \pi[$. On indique généralement un sens de rotation arbitraire. L'une des spécificités des mesures angulaires est l'absence de la notion d'ordre sur le cercle à cause de la périodicité, l'angle π est supérieur à zéro mais ce même angle s'écrit $-\pi$ et devient ainsi inférieur à zéro. Ceci est l'une des raisons pour lesquelles les statistiques dans le domaine linéaires ne sont pas adaptées au domaine angulaire.

Prenons à titre d'exemple la moyenne de deux angles et la distance entre deux angles. Sur la figure 3.7, la moyenne des deux angles 20° et 340° est 180° alors qu'elle devait être 0° puisque les deux angles sont symétriques. De même, la distance entre ces deux angles calculée par une simple soustraction est de 320° alors qu'elle est de 40° .

C'est pour cette raison qu'il est nécessaire de recourir à d'autres modèles statistiques pour traiter les variables circulaires qui prennent en considération la nature périodique indépendamment de la référence zéro.

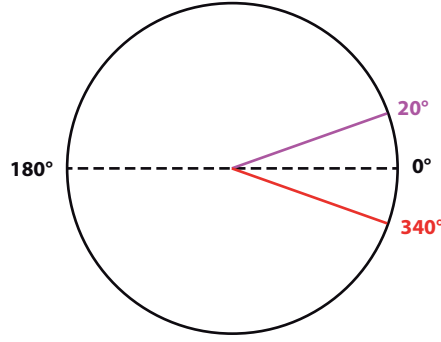
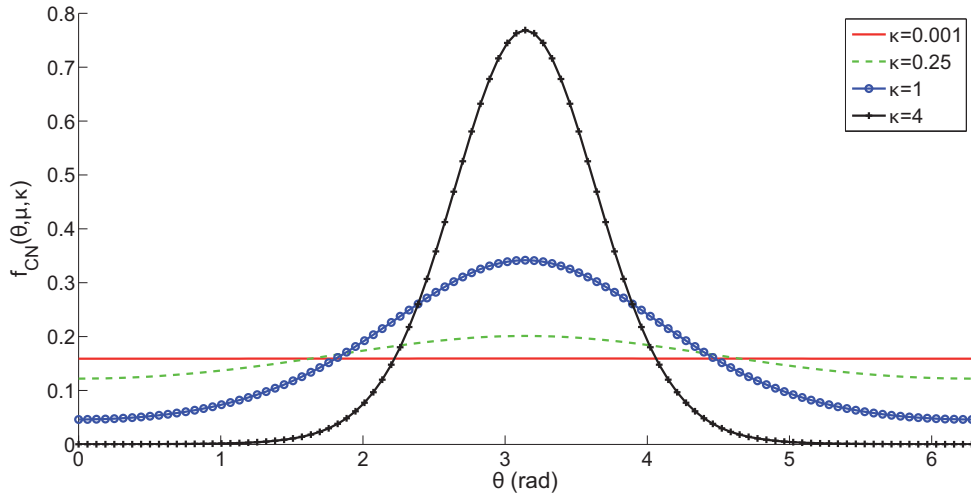


FIGURE 3.7 – Statistiques sur un cercle

La distribution de von Mises ou distribution circulaire normale est l'équivalent circulaire de la distribution normale dans le domaine linéaire. Elle est 2π -périodique. Sa densité de probabilité $f_{CN}(\theta; \mu, \kappa)$ de la variable circulaire θ , de moyenne μ et de facteur de concentration κ s'écrit :

$$f_{CN}(\theta; \mu, \kappa) = \frac{1}{2\pi I_0(\kappa)} e^{\kappa \cos(\theta - \mu)} \quad (3.15)$$


FIGURE 3.8 – Distribution de von Mises pour différents κ et $\mu = \pi$

Le facteur de concentration κ est l'équivalent de l'inverse de la variance pour une loi normale. Il varie entre 0 et l'infini. Plus il est grand, plus les réalisations sont concentrées autour de la moyenne μ . Pour de grandes valeurs de κ , la distribution de von Mises tend vers une loi normale de variance $1/\kappa$. Lorsque κ tend vers zéro, la distribution de von Mises tend vers une distribution uniforme entre 0 et 2π comme illustré sur la figure 3.8.

I_0 est la fonction de Bessel modifiée de première espèce et d'ordre zéro. Il s'agit d'une constante de normalisation [33] :

$$I_0(\kappa) = \frac{1}{2\pi} \int_0^{2\pi} e^{\kappa \cos \theta} d\theta = \sum_{r=0}^{\infty} \left(\frac{\kappa}{2}\right)^{2r} \left(\frac{1}{r!}\right)^2 \quad (3.16)$$

La distribution de von Mises sera utilisée pour représenter la direction du véhicule dans le Map matching particulière décrit dans ce chapitre ainsi que dans la description du filtre circulaire multi-modèles présenté dans le chapitre suivant.

3.3.3 Implémentation particulière

On dénote les coordonnées du dernier point GPS disponible par $(x_p, y_p)^T$ obtenu avec une précision définie respectivement par les variances de l'erreur horizontale et verticale v_x et v_y . Dans l'étape d'initialisation, on note par v le segment associé à cette position GPS et par V l'ensemble des segments qui y sont connectés dans le sens du mouvement du véhicule. On note k les instants associés à l'échantillonnage du signal observé. L'algorithme de Map matching particulière se déroule comme suit :

1. Initialisation :

Pour estimer l'identité du segment, on génère N particules $x_0^i = (X_0^i, Y_0^i)^T$, $i = 1, \dots, N$. X_0^i, Y_0^i qui sont tirées d'une distribution normale de moyenne m , la projection de x_p, y_p sur le segment courant r_0 de direction $\alpha(r_0)$, et de variance $\frac{1}{2} \left(\frac{v_x}{\cos(\alpha(r_0))}, \frac{v_y}{\sin(\alpha(r_0))} \right)$. Les poids sont initialisés à :

$$w_0^i = \frac{1}{2\pi \sqrt{v_x v_y}} \exp \left(-\frac{(X_0^i - x_p)^2}{2v_x} - \frac{(Y_0^i - y_p)^2}{2v_y} \right) \quad (3.17)$$

$$(3.18)$$

2. Etape de prédiction :

Les particules de position sont propagées sur les segments du réseau de routes à partir des équations suivantes :

$$X_k^i = X_{k-1}^i + v_{k-1} \Delta t \cos(\alpha(R_{k-1}^i)) + \nu_k^i \cos(\alpha(R_{k-1}^i)) \quad (3.19)$$

$$Y_k^i = Y_{k-1}^i + v_{k-1} \Delta t \sin(\alpha(R_{k-1}^i)) + \nu_k^i \sin(\alpha(R_{k-1}^i)) \quad (3.20)$$

Où ν_k^i est distribué suivant un bruit aléatoire qui modélise l'incertitude sur la commande v_{k-1} . ν_k^i suit une distribution uniforme de moyenne zéro et de variance ν_Q . R_{k-1}^i définit le

segment de route sur lequel se trouve la particule i à l'instant k . Au début du segment les particules sont distribuées suivant une loi gaussienne et elles sont propagées par la commande $v_{k-1} \Delta t$. Les particules ne sont pas propagées suivant le bruit d'état mais suivant le bruit de commande car elles ne sont pas corrigées en position. En effet c'est l'identité du segment sur lequel la particule est propagée qui est corrigé grâce à la différence entre la direction du segment et la direction observée. En revanche les particules sont propagées sur le segment avec la commande. On suppose ici que l'erreur sur la valeur de la commande est principalement un biais. La valeur de ce biais est distribué uniformément sur toutes les valeurs possibles définies par la variance de la loi.

Les particules utilisées pour identifier la route sont propagées par la formule suivante :

$$R_k^i = f(R_{k-1}^i, X_k^i, Y_k^i) \quad (3.21)$$

La fonction $f(\dots)$ est une fonction non linéaire dans laquelle on décide, à partir des coordonnées X_k^i, Y_k^i , que la particule est sortie du segment courant R_{k-1}^i . Lorsque celle-ci est sortie, on tire, suivant une loi uniforme parmi l'ensemble des routes connectées au segment courant, la nouvelle route sur laquelle se trouve la particule. Donc, chacune des particules est associée à une position sur le segment (X_k^i, Y_k^i) , et est associée à l'identité d'un segment R_k^i .

3. Etape de mise à jour :

Les poids utilisés pour estimer l'identité de la route courante sont calculés en fonction de la direction estimée :

$$\tilde{w}_k^i = w_{k-1}^i f_{CN}(\hat{\theta}_k; \alpha(R_k^i), \kappa_{R_p}) \quad (3.22)$$

où la distribution de von Mises f_{CN} est symétrique et uni-modale sur une période. κ_{R_p} indique notre tolérance à l'existence de particules sur des segments peu probables. Les segments seront d'autant moins probables que leur direction est différente de $\hat{\theta}_k$, la direction estimée par le filtre circulaire. Les poids sont calculés puis normalisés :

$$w_k^i = \frac{\tilde{w}_k^i}{\sum_{j=1}^N \tilde{w}_k^j} \quad (3.23)$$

4. Estimation :

Les coordonnées de la position du véhicule (\hat{x}_k, \hat{y}_k) sont d'abord pré-définies à partir du poids des particules par la formule suivante :

$$\hat{x}p_k = \sum_{i=1}^N w_k^i X_k^i \quad (3.24)$$

$$\hat{y}p_k = \sum_{i=1}^N w_k^i Y_k^i \quad (3.25)$$

$$(3.26)$$

Elles sont ensuite projetées sur le segment le plus proche \hat{r}_k , à la position estimée (\hat{x}_k, \hat{y}_k) . \hat{r}_k est donc l'estimation de l'identité du segment de route.

5. Ré-échantillonnage :

Le ré-échantillonnage utilisé est le Ré-échantillonnage d'importance séquentielle SIR (*Sequential Importance Resampling*) [4] avec un nombre effectif de particules défini comme suit :

$$N_{eff} = \frac{1}{\sum_{i=1}^N w_k^i{}^2} \quad (3.27)$$

Par cette méthode, les particules sont redistribuées suivant la distribution de l'état estimé sachant les mesures et conservent la position (X_k^i, Y_k^i) et l'identité du segment R_k^i sur lequel elles se trouvent.

Algorithme du filtre particulaire

Voici l'algorithme du filtre particulaire à l'instant k qui résume les différentes étapes détaillées précédemment :

- POUR $i = 1 : N$
 - Tirer $(X_k^{(i)}, Y_k^{(i)}) \sim f_1((x_k, y_k) | (X_{0:k-1}^{(i)}, Y_{0:k-1}^{(i)}))$
 - Tirer¹ $R_k^{(i)} \sim f_2(r_k | (X_k^{(i)}, Y_k^{(i)}))$
 - Calculer le poids de chaque particule : $w_k^{(i)} = w_{k-1}^{(i)} g(\hat{\theta}_k | \alpha(R_k^i))$
- FIN POUR
- Calculer la somme des poids : $S = \sum_{i=1}^N w_k^{(i)}$
- POUR $i = 1 : N$
 - Normaliser : $w_k^{(i)} = w_k^{(i)} / S$

1. En fonction des valeurs de $(X_k^{(i)}, Y_k^{(i)})$ sur l'ensemble $\{r_k\}$ des segments connectés au segment courant r_k .

- FIN POUR
- Calculer $\widehat{N_{eff}} = 1 / \sum_{i=1}^N (w_k^{(i)})^2$
- SI $\widehat{N_{eff}} < N_T$ ALORS
 - Ré-échantillonner les particules avec un algorithme de ré-échantillonnage
 - Réinitialiser tous les poids : $w_k^{(i)} = 1/N$
- FIN SI

Où $f_1(\dots)$ et $f_2(\dots)$ sont des lois uniformes et $g(\dots)$ une loi de von Mises.

3.3.4 Fonctionnement du filtre

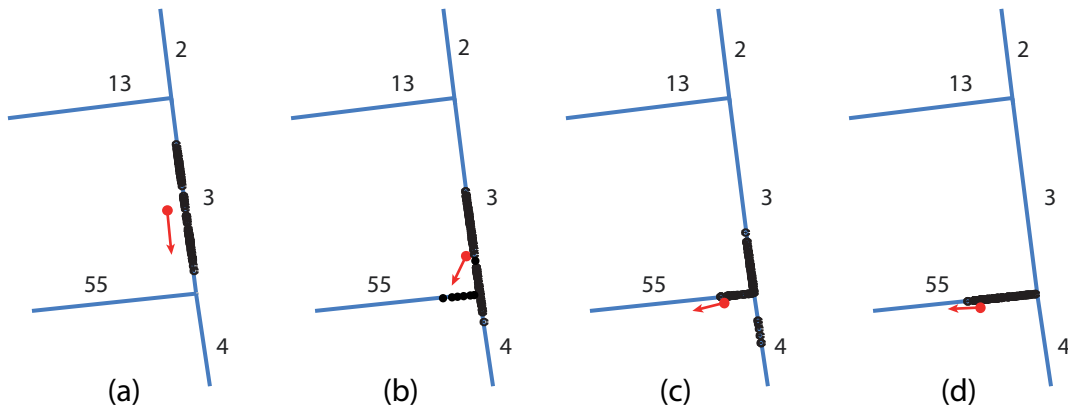


FIGURE 3.9 – Comportement du filtre particulaire dans un virage

Dans l'exemple illustré par la figure 3.9, on observe le comportement du filtre particulaire lorsque le véhicule arrive à un virage. La position réelle du véhicule (supposée inconnue) ainsi que sa direction sont représentées par le point et le vecteur rouge respectivement. Durant l'étape (a), toutes les particules sont situées sur le segment 3 puisque aucune particule n'est arrivée en fin de segment. A l'étape (b), le véhicule commence une manœuvre de rotation vers le segment 55. Quelques particules arrivées en fin de segment 3 sont placées sur les segments 4 et 55 qui constituent un chemin probable, vue la direction du mouvement. Les particules du segment 55 ont un faible poids car la direction du mouvement ne correspond pas avec la direction du segment. A l'étape (c), le véhicule finit sa rotation, les particules du segment 55 ont alors un fort poids car elles sont alignées dans la direction de mouvement du véhicule. Le poids des particules du segment 4 diminue (elles diminuent en nombre après l'étape de Ré-échantillonnage car elles sont redistribuées sur le segment le plus probable), à cause de l'écart croissant entre la direction du segment 4 et la direction du véhicule. Enfin, à l'étape (d), le véhicule continue son mouvement sur le segment 55. Toutes les particules sont maintenant sur ce segment car sa direction est la plus proche de la direction du véhicule et aucune particule n'a atteint la fin du

segment.

3.4 Pré-filtrage circulaire

3.4.1 Architecture du système de préfiltrage

Pour améliorer les performances du filtre particulaire de Map matching, on filtre préalablement les observations de la direction. En effet, l'estimation de la direction étant indépendante de l'estimation de la position, on considère deux filtres distincts fonctionnant l'un à la suite de l'autre. On représente figure 3.10 l'architecture du système proposé. Dans cette architecture, les filtres de prétraitement des observations de direction ont été proposés dans [76] et sont décrits succinctement dans la suite du paragraphe. L'objectif est ici d'estimer θ_k la direction et $\tilde{\theta}_k$ sa variation angulaire.

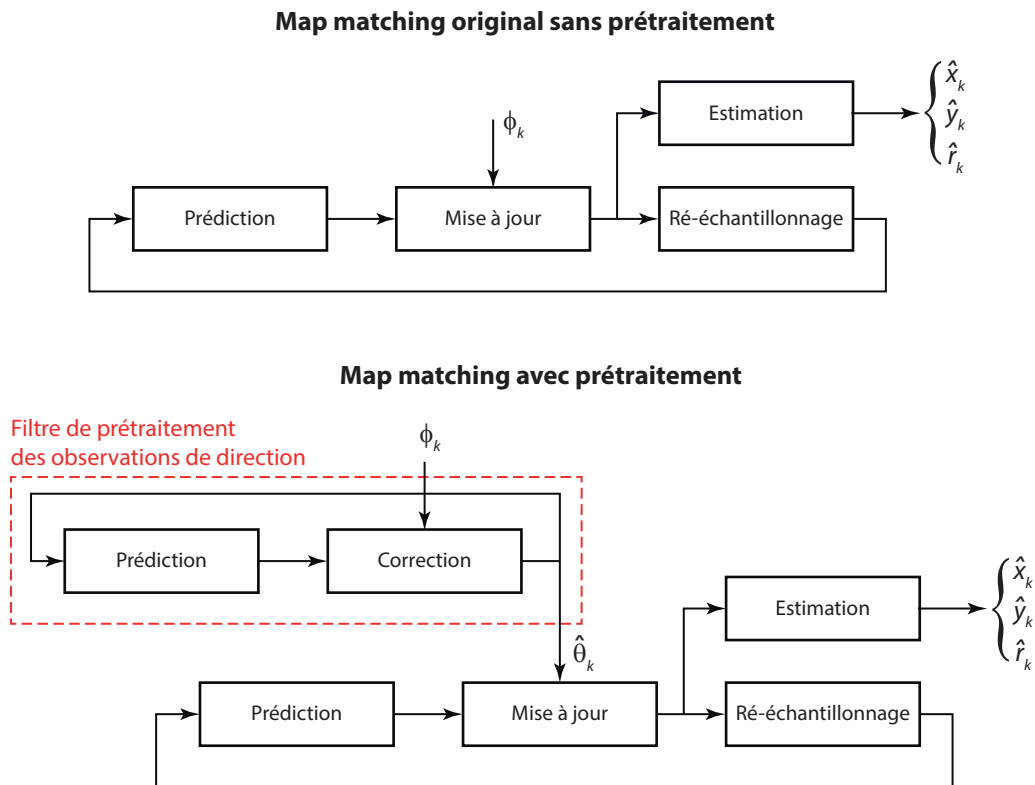


FIGURE 3.10 – Architecture du filtre de Map matching

3.4.2 Pré-filtrage d'état circulaire

Le filtre circulaire récursif est défini dans le cadre bayésien pour estimer les grandeurs circulaires. Soit un système à temps discret de loi initiale $f(\theta_0)$, de loi de transition $f(\theta_k|\theta_{k-1}, \kappa_Q)$ et de fonction de vraisemblance $f(\phi_k|\theta_k, \kappa_R)$, avec ϕ_k la mesure angulaire. Le système est régi par les équations d'état et de mesure suivantes :

$$\theta_k = \theta_{k-1} + \tilde{\theta}_{k-1} + v_{k-1} \quad (3.28)$$

$$\tilde{\theta}_k = \tilde{\theta}_{k-1} + \tilde{v}_{k-1} \quad (3.29)$$

$$\phi_k = \theta_k + \omega_k$$

v_{k-1} , \tilde{v}_{k-1} et ω_k sont les bruits de modèle et de mesure respectivement suivant une distribution de von Mises centrée sur zéro et de paramètre de concentration κ_Q , $\kappa_{\tilde{Q}}$ et κ_R respectivement.

L'algorithme du filtre circulaire se déroule, comme le filtre de Kalman, selon un schéma en deux étapes : prédiction (se reportant à l'équation d'état) et correction (se reportant à l'équation de mesure). On décrit ici l'implémentation du filtre dans le cas où l'on dispose uniquement des observations de direction, contrairement au cas décrit dans [76].

Étape de prédiction

La loi *a posteriori* à l'instant $k-1$ suit une distribution de von Mises :

$$\theta_{k-1}|\phi_{1:k-1} \sim \mathcal{CN}(\theta_{k-1}|\hat{\theta}_{k-1}, P_{\theta_{k-1}}) \quad (3.30)$$

$\hat{\theta}_{k-1}$ et $P_{\theta_{k-1}}$ sont respectivement l'estimée de la direction et le paramètre de concentration de θ_{k-1} .

On démontre dans [6, 76] que la distribution de l'état prédit $\theta_{k|k-1}$, définie par l'équation (3.28), peut être approximée par :

$$\mathcal{CN}(\theta_k|\hat{\theta}_{k|k-1}, P_{\theta_{k|k-1}}) \approx \mathcal{WN}(\theta_k|\hat{\theta}_{k-1}, A(\kappa_Q)A(P_{\theta_{k-1}})) \quad (3.31)$$

où $\mathcal{WN}(\mu, \rho)$ est La distribution Wrapped Normal. On montre que, $\rho = A(\kappa)$ et $\kappa = A^{-1}(\rho)$. D'où l'état prédit et son paramètre de concentration :

$$\hat{\theta}_{k|k-1} = \hat{\theta}_{k-1} + \hat{\tilde{\theta}}_{k-1} \quad (3.32)$$

$$\hat{\tilde{\theta}}_{k|k-1} = \hat{\tilde{\theta}}_{k-1} \quad (3.33)$$

$$P_{\theta_{k|k-1}} = A^{-1} \left(A(\kappa_Q) A(P_{\theta_{k-1}}) A(P_{\tilde{\theta}_{k-1}}) \right) \quad (3.34)$$

$$P_{\tilde{\theta}_{k|k-1}} = A^{-1} \left(A(\kappa_{\tilde{Q}}) A(P_{\tilde{\theta}_{k-1}}) \right) \quad (3.35)$$

Plusieurs méthodes de calcul de $A^{-1}(\cdot)$ sont discutées dans [76, 27] comme le développement limité, la méthode de Newton et la méthode de Runge-Kutta.

Étape de correction

Dans cette étape, on calcule l'estimée $\hat{\theta}_k$ et sa variance P_{θ_k} en utilisant la loi *a posteriori* $f(\theta_k|\phi_{1:k})$ donnée par :

$$f(\theta_k|\phi_{1:k}) \propto f(\theta_k|\phi_{1:k-1}) f(\phi_k|\theta_k) \quad (3.36)$$

$f(\theta_k|\phi_{1:k-1})$ étant la distribution *a priori* fournie par l'étape de prédiction, et $f(\phi_k|\theta_k)$ la vraisemblance indiquant le degré de similarité entre la mesure et l'état.

On suppose dans le cas circulaire que les distributions *a priori* et *a posteriori* sont des distributions de von Mises, d'où :

$$\mathcal{CN}(\theta_k|\hat{\theta}_k, P_{\theta_k}) \propto \mathcal{CN}(\theta_k|\hat{\theta}_{k|k-1}, P_{\theta_{k|k-1}}) \mathcal{CN}(\phi_k|\theta_k, \kappa_R) \quad (3.37)$$

On démontre alors dans [76] que l'estimée et le facteur de concentration de la direction et de la variation d'angle s'écrivent :

$$C_{\theta_k} = \alpha_{11} \cos \hat{\theta}_{k|k-1} + \alpha_{12} \cos \phi_k \quad (3.38)$$

$$S_{\theta_k} = \alpha_{11} \sin \hat{\theta}_{k|k-1} + \alpha_{12} \sin \phi_k \quad (3.39)$$

avec

$$\alpha_{11} = P_{\theta_k|k-1} \quad (3.40)$$

$$\alpha_{12} = \kappa_R \quad (3.41)$$

$$(3.42)$$

Et :

$$C_{\hat{\theta}_k} = \alpha_{21} \cos \hat{\theta}_{k|k-1} + \alpha_{22} \cos(\phi_k - \hat{\theta}_{k-1}) \quad (3.43)$$

$$S_{\hat{\theta}_k} = \alpha_{21} \sin \hat{\theta}_{k|k-1} + \alpha_{22} \sin(\phi_k - \hat{\theta}_{k-1}) \quad (3.44)$$

avec

$$\alpha_{21} = P_{\hat{\theta}_{k|k-1}} \quad (3.45)$$

$$\alpha_{22} = A^{-1}(A(P_{\theta_{k-1}})A(\kappa_R)) \quad (3.46)$$

Finalement :

$$P_{\theta_k} = \sqrt{C_{\hat{\theta}_k}^2 + S_{\hat{\theta}_k}^2} \quad (3.47)$$

$$P_{\hat{\theta}_k} = \sqrt{C_{\hat{\theta}_k}^2 + S_{\hat{\theta}_k}^2} \quad (3.48)$$

$$\hat{\theta}_k = \text{Arctan}^*(S_{\theta_k}, C_{\theta_k}) \quad (3.49)$$

$$\hat{\hat{\theta}}_k = \text{Arctan}^*(S_{\hat{\theta}_k}, C_{\hat{\theta}_k}) \quad (3.50)$$

\arctan^* est la fonction réciproque "*quadrant specific*" de la tangente [33], connue aussi comme *atan2*. Elle prend en considération le signe de ses deux paramètres pour retourner un angle entre $-\pi$ et π dans l'un des quatre quadrants du cercle unitaire :

$$\arctan^*(S, C) = \begin{cases} \arctan(S/C) & \text{si } C > 0 \\ \arctan(S/C) + \pi & \text{si } C < 0, S \geq 0 \\ \arctan(S/C) - \pi & \text{si } C < 0, S < 0 \\ \pi/2 & \text{si } C = 0, S > 0 \\ -\pi/2 & \text{si } C = 0, S < 0 \\ \text{indéfini} & \text{si } C = 0, S = 0 \end{cases} \quad (3.51)$$

3.4.3 Pré-filtrage particulière circulaire

1. Initialisation :

Pour l'estimation de la direction on génère N particules, notées Θ_0^i . Les particules sur l'angle Θ_0^i suivent une distribution de von Mises de moyenne ϕ_0 et de paramètre de concentration κ_R . Pour l'estimation de la variation angulaire, on génère N particules, notées $\tilde{\Theta}_0^i$ suivant une distribution de von Mises de moyenne 0, et de paramètre de concentration $\kappa_{\tilde{R}}$. Chaque particule est affectée d'un poids initial $w_0^i = 1/N$.

2. Etape de prédiction :

Les particules utilisées pour l'estimation de la direction sont propagées sur la composante angulaire par la formule suivante :

$$\tilde{\Theta}_k^i = \tilde{\Theta}_{k-1}^i \quad (3.52)$$

$$\Theta_k^i = \Theta_{k-1}^i + \tilde{\Theta}_{k-1}^i + \nu_k^i \quad (3.53)$$

où ν_k^i suit une distribution de von Mises de moyenne 0, de variance ν_Q .

3. Etape de mise à jour :

Les poids utilisés pour estimer la direction sont calculés en fonction des mesures de direction :

$$\tilde{w}_k^i = w_{k-1}^i f_{CN}(\phi_k; \Theta_k^i, \kappa_R) \quad (3.54)$$

La distribution de von Mises f_{CN} est symétrique, uni-modale sur une période. ϕ_k définit la mesure de la direction du véhicule, Θ_k^i la direction de la particule, et κ_R le facteur de concentration. Les poids sont calculés puis normalisés :

$$w_k^i = \frac{\tilde{w}_k^i}{\sum_{j=1}^N \tilde{w}_k^j} \quad (3.55)$$

4. Estimation :

La direction est estimée avec la formule suivante :

$$\hat{\theta}_k = \arg \left(\sum_{i=1}^N w_k^i \exp(j\Theta_k^i) \right) \quad (3.56)$$

5. Ré-échantillonnage :

Dans le cas de l'estimation de la direction, le ré-échantillonnage choisi ici est le Ré-échantillonnage d'importance séquentielle SIR (*Sequential Importance Resampling*) [4], avec un nombre effectif de particules défini comme suit :

$$N_{eff} = \frac{1}{\sum_{i=1}^N w_k^i{}^2} \quad (3.57)$$

3.5 Évaluation des méthodes

3.5.1 Introduction

Dans cette expérimentation, on évalue les méthodes proposées lorsque le système GPS est masqué. On considère alors comme point initial le dernier point GPS avant masquage et les mesures de vitesse et direction fournies par les capteurs comme observations. On dispose aussi d'une carte numérique de la ville de Calais en France. Cette carte, au format vectoriel, est issue du site www.openstreetmap.org. On suppose que le véhicule se trouve sur une des routes référencées sur la carte.

On souhaite évaluer les méthodes proposées en termes de précision de l'estimation et d'identification de la route. Concernant l'estimation, on évalue la précision des méthodes proposées pour la direction du véhicule et le positionnement de celui-ci. Concernant l'identification de la route, on détermine la probabilité d'identifier celle-ci correctement.

Pour évaluer la précision de l'estimation, on considère l'intersection en Y présentée figure 3.11.(a). La trajectoire du véhicule débute sur la route en bas à droite de la figure et se poursuit à la bifurcation sur le segment de gauche. Pour évaluer la probabilité d'identifier correctement la route, on considère à la fois la configuration en Y de la figure 3.11.(a) et l'expérimentation définie dans un contexte réel de la figure 3.11.(b). Dans ce dernier cas, la zone et la durée de masquage des informations GPS sont choisies aléatoirement. La distance parcourue par le véhicule est de 580m, le trajet étant d'une durée de 125s (125 points GPS à 1Hz) avec un arrêt de 15s à un feu rouge.

Pour évaluer les méthodes en fonction du rapport signal à bruit, on comparera leurs performances pour différents angles entre les segments de la jonction en Y. Dans ce contexte, les bruits de mesures capteurs sont fixés. L'erreur sur la mesure de direction est distribuée suivant une loi de von Mises centrée de paramètre de concentration $\kappa = 30$. La vitesse du véhicule

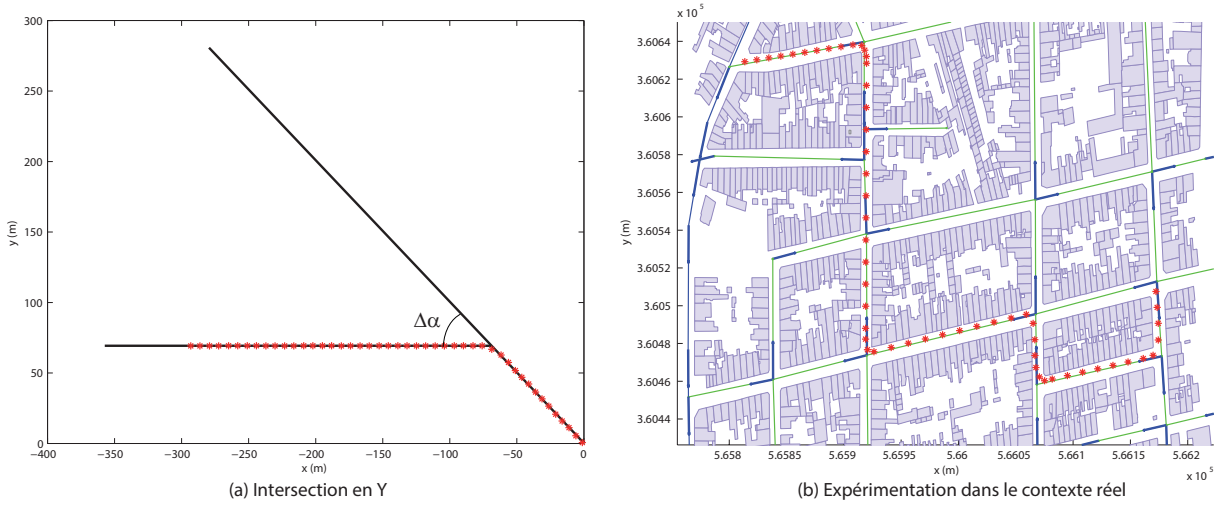


FIGURE 3.11 – Les trajectoires d'évaluation

à l'arrivée à la bifurcation est de 10 km/h ($\approx 2 \text{ m/s}$). L'erreur sur la mesure de vitesse est supposée gaussienne et biaisée. La variance de la distribution est fixée à 1 m/s et le biais est supposé distribué suivant une loi uniforme entre 0.5 m/s et -0.5 m/s .

3.5.2 Estimation de la direction

Dans notre expérimentation, on compare la méthode proposée avec le Map matching topologique décrit dans [70] (voir section 1.4.2). Dans notre approche, on utilise un filtre pour pré-traiter les observations de la direction du véhicule. Nous avons choisi le filtre circulaire décrit dans [79] plutôt que le filtre de Kalman classique ou le filtre particulier pour estimer la direction. On représente figure 3.12.(a) la direction estimée par un filtre de Kalman, le filtre circulaire et le filtre particulier dans le cas de la jonction en Y testée. On représente figure 3.12.(b) la distance circulaire entre la direction réelle θ_k et la direction estimée $\hat{\theta}_k$. La distance circulaire varie entre 0 et 2. Elle est définie par :

$$dc_k = 1 - \cos(\theta_k - \hat{\theta}_k) \quad (3.58)$$

On peut constater figure 3.12.(a) et figure 3.12.(b) que l'estimation fournie par le filtre de Kalman est erronée pour des angles proches de π , car dans ce cas la distance circulaire sur l'erreur d'estimation est forte. En effet, dans ce cas les transitions entre $-\pi$ et π causées par le bruit sont "vues" par le filtre de Kalman comme des sauts de 2π .

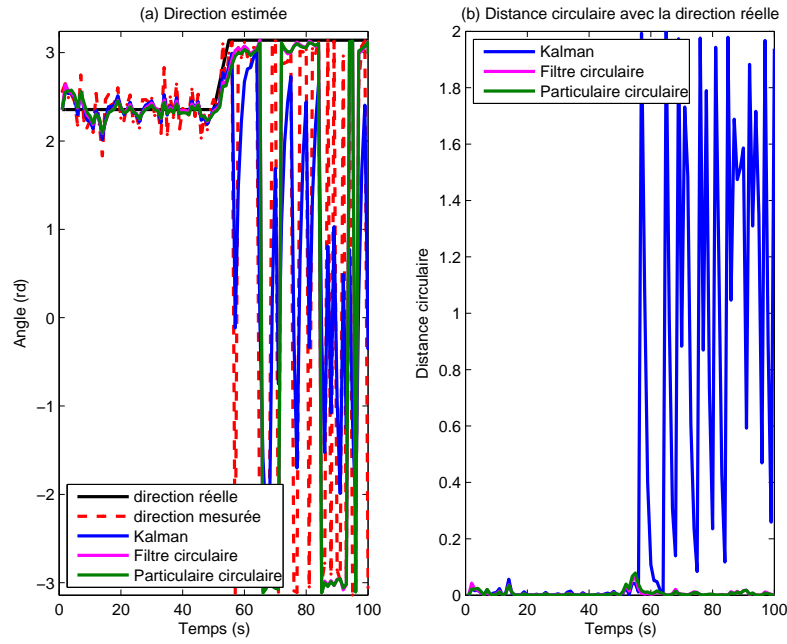


FIGURE 3.12 – Estimation de la direction

Pour évaluer l'estimation de la direction, on calcule pour chaque instant la dispersion circulaire. La dispersion circulaire est décrite par la moyenne des distances circulaires calculées à chaque instant avec les directions estimées sur 100 réalisations de bruit. On reporte figure 3.13.(a) la dispersion circulaire en fonction du temps et figure 3.13.(b) la dispersion circulaire des mesures en fonction des réalisations.

La dispersion circulaire moyenne (calculée sur les 100 réalisations) est de 0.017 pour la mesure, 0.481 pour le filtre de Kalman, 0.006 pour le filtre particulaire circulaire et 0.005 pour le filtre circulaire. Le filtre de Kalman fournit donc la moins précise des estimations de la direction. On observe une pointe sur la courbe de dispersion circulaire figure 3.13.(a) pendant la manœuvre. On montre figures 3.13 que les filtres circulaires fournissent des résultats avec une précision similaire. Cependant, on utilisera dans la suite le filtre circulaire pour estimer la direction, car son coût de calcul est plus faible que le filtre particulaire.

3.5.3 Estimation de la position

Dans notre expérimentation, on compare la méthode proposée, soit le Map matching particulaire (noté MAP-SIR) avec le Map matching topologique proposé en [70] (noté MAP-TOP). Dans l'évaluation on compare le filtre proposé MAP-SIR, avec et sans le prétraitement de l'angle mesuré, avec le filtre de référence MAP-TOP.

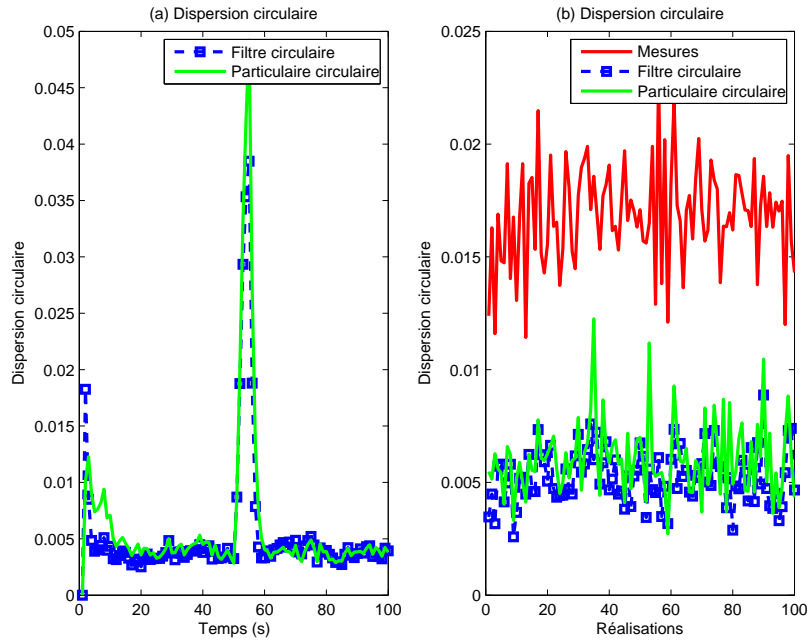


FIGURE 3.13 – Estimation de la direction

Pour évaluer l'estimation de la position, on calcule la distance moyenne en fonction du temps entre la position réelle et la position estimée par les différentes méthodes de Map matching. On calcule aussi pour chaque réalisation, l'erreur moyenne sur les positions estimées d'une trajectoire. La position moyenne et l'erreur sont obtenues à partir de 1 000 estimations de la trajectoire.

On reporte figure 3.14 l'erreur sur la position obtenue avec les filtres de map matching, lorsque la mesure de la direction n'est pas préalablement filtrée, et lorsqu'elle est filtrée avec le filtre circulaire et pour une bifurcation de $\Delta\alpha = 45^\circ$.

Les résultats obtenus figure 3.14 montrent que l'on améliore le positionnement par l'algorithme MAP-SIR proposé, car l'erreur est plus faible. De plus, on définit l'erreur de positionnement globale qui est la distance moyenne entre les positions réelles et les positions estimées pour 1 000 estimations de la trajectoire en Y. Cette erreur calculée pour la trajectoire en Y est de 11.2 m pour l'algorithme MAP-TOP, de 9.1 m pour l'algorithme MAP-SIR sans prétraitement de la direction, et de 8.1 m pour l'algorithme MAP-SIR avec pré-traitement de la direction par le filtre circulaire. Cependant, on peut observer figure 3.14 que la distance moyenne entre la position réelle et la position estimée par l'algorithme MAP-SIR augmente lorsqu'on se rapproche de la bifurcation (à $t=50s$) et devient supérieure à celle estimée par l'algorithme MAP-TOP.

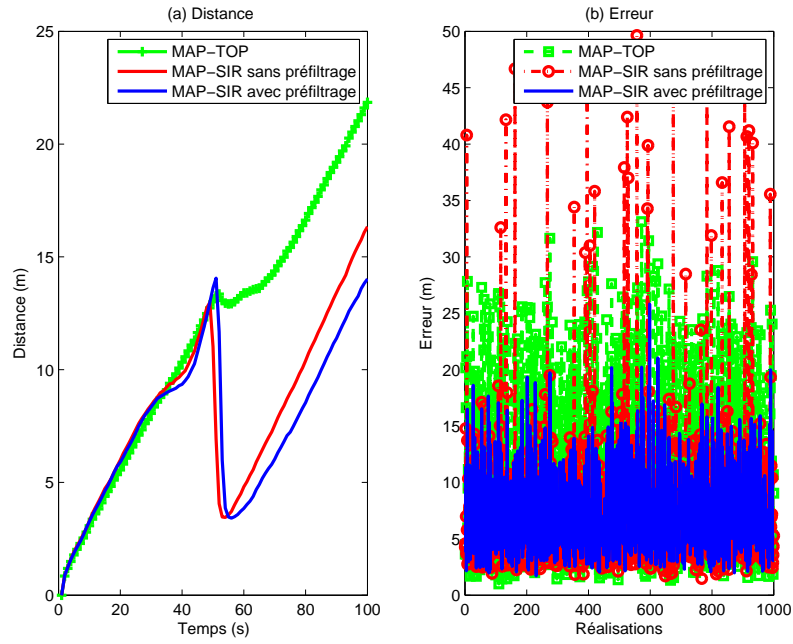


FIGURE 3.14 – Erreur sur la position avec et sans pré-filtrage

Cette augmentation est due au retard concernant l'identification du segment courant, lié au lissage de la direction. Dans ce cas, la bifurcation est détectée en retard, les particules continuent à se propager sur le segment précédent et s'éloignent de la bifurcation.

On notera que l'algorithme proposé permet de ramener l'erreur de position à 3.5 m après la bifurcation. Cette diminution de l'erreur correspond au recalage de la position avec la direction. L'erreur de 3.5 m est associée à un retard dû à la détection d'environ deux secondes pour un véhicule circulant à 2 m/s et une fréquence d'échantillonnage d'une seconde. Les performances de l'algorithme pourront être améliorées en augmentant la fréquence d'échantillonnage des données capteurs.

Pour évaluer la robustesse de la méthode proposée, on calcule l'erreur de positionnement en fonction de $\Delta\alpha$, l'angle entre les routes de directions α . On reporte dans le tableau 3.1 cette erreur en mètres en fonction de $\Delta\alpha$.

$\Delta\alpha$ (°)	45	34	22	11
MAP-TOP	11.2	12.0	14.4	14.8
MAP-SIR	8.1	8.0	8.7	9.5

TABLE 3.1 – Erreur de positionnement globale en fonction de l'angle entre les routes.

Les résultats reportés dans le tableau 3.1 montrent que le filtre proposé est plus robuste au bruit que l'approche classique. En effet, l'erreur de positionnement obtenue avec le filtre proposé est toujours inférieure ou égale à l'erreur obtenue avec le filtre classique lorsque le rapport signal à bruit diminue avec la différence de directions $\Delta\alpha$. On reporte dans le tableau 3.2 les paramètres des filtres pour les différentes expérimentations.

$\Delta\alpha$ (°)	45	34	22	11
Filtre Circulaire				
κ_R	140	140	140	140
κ_Q	1e3	1e3	1e3	1e3
$\kappa_{\tilde{Q}}$	1e4	1e4	1e4	1e4
MAP-SIR				
N	200	200	200	200
κ_{R_p}	100	100	100	100
ν_Q	9	9	9	9
MAP-TOP				
Seuil $\Delta\beta'$ (°)	45	45	45	45
Seuil α (°)	90	90	90	90

TABLE 3.2 – Paramètres des filtres.

3.5.4 Identification de la route

Évaluation dans le cas de la trajectoire Y

On mesure ici les performances du Map matching particulière sur des données synthétiques, et dans le cas de la trajectoire Y du paragraphe précédent. On souhaite évaluer la probabilité d'identifier correctement le segment sur lequel se trouve le véhicule. On reporte figure 3.15 cette probabilité en fonction de la position du véhicule sur la trajectoire. Cette probabilité est évaluée à partir de 1 000 réalisations de la trajectoire.

On constate que l'algorithme proposé permet d'identifier le changement de route à l'instant de la jonction plus rapidement que l'algorithme classique MAP-TOP. En effet, la probabilité de définir correctement l'identité de la route est plus forte aux instants proches de la jonction. Pour évaluer les performances globales de l'identification dans le cas du Y, on calcule la probabilité d'identifier le bon segment sur l'ensemble de la trajectoire. On reporte dans le tableau 3.3 cette probabilité en fonction de l'angle entre les routes de la jonction.

Les résultats décrits dans le tableau 3.3 montrent que la méthode proposée est plus fiable que l'approche MAP-TOP, car elle offre une probabilité d'identification de la route sur laquelle

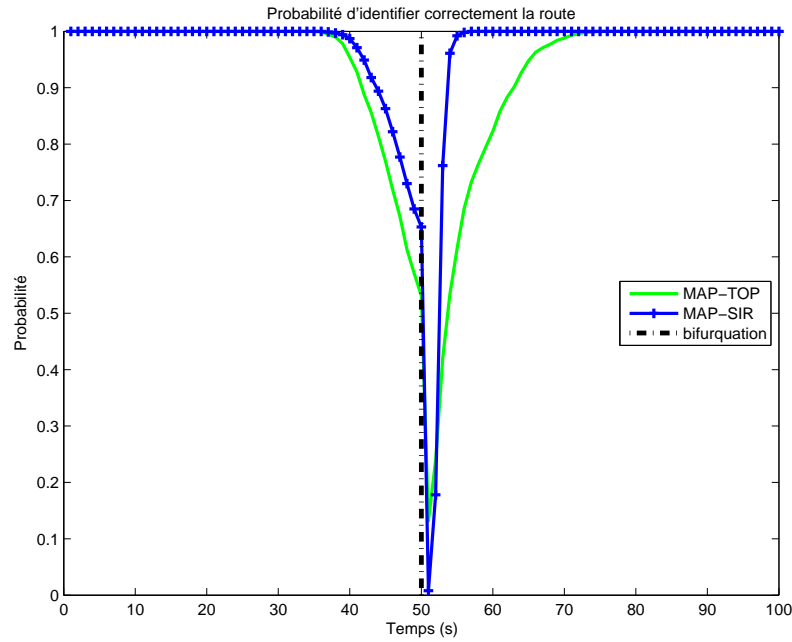


FIGURE 3.15 – Probabilité d’identifier correctement le segment de route

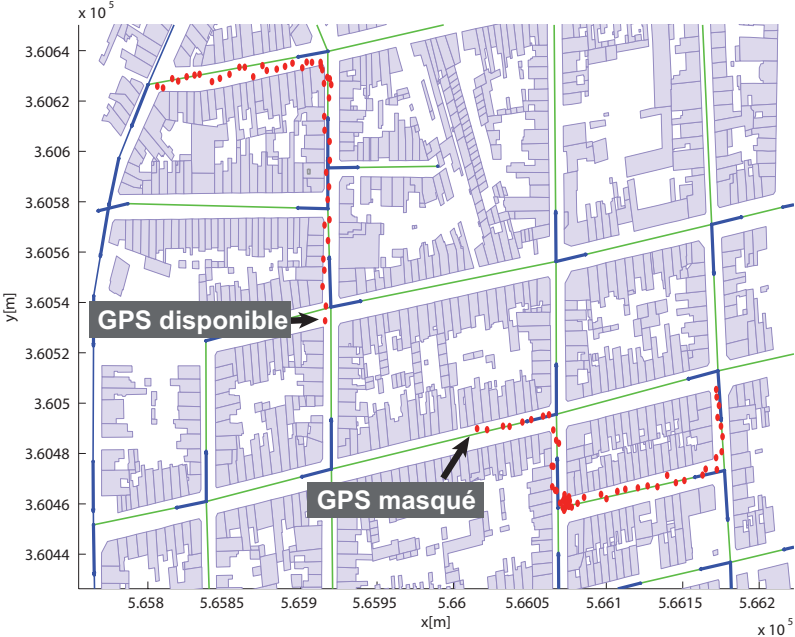
$\Delta\alpha$ (°)	45	34	22	11
MAP-TOP	0.915	0.895	0.801	0.772
MAP-SIR	0.943	0.946	0.937	0.926

TABLE 3.3 – Probabilité d’identification en fonction de l’angle entre les routes.

se trouve le véhicule plus grande. On montre aussi que la méthode est plus robuste, car la probabilité reste supérieure lorsque le rapport signal à bruit diminue (l’angle $\Delta\alpha$ diminue).

Évaluation dans un contexte réel

On mesure ici les performances du Map matching particulière sur des données synthétiques et dans un contexte réel. On simule ici des mesures fournies par les capteurs d’un véhicule équipé d’un récepteur GPS, un odomètre pour la mesure de vitesse et un magnétomètre pour la mesure de direction. Les mesures sont simulées dans le cas d’une trajectoire reconstruite à l’aide de données réelles. On considère la trajectoire illustrée sur la figure 3.16 et une fréquence des mesures des capteurs proprioceptifs et GPS de 1 Hz. Dans cette expérimentation on considère une erreur capteur sur les mesures dont la statistique est obtenue à partir des données réelles. Cette erreur possède une variance de 5%, ainsi qu’un biais de 5% pour la mesure de vitesse et un paramètre de concentration de 30 pour la distribution de von Mises qui modélise l’erreur sur la direction. L’erreur GPS est définie comme la variance de la distance entre la position mesurée et la position réelle.



Dans une première expérimentation, on compare les performances de l'algorithme topologique MAP-TOP avec le filtre particulaire MAP-SIR lorsque le masquage est total. On considère tous les points GPS masqués, sauf le premier pour MAP-SIR, et tous les points GPS disponibles pour MAP-TOP avec des précisions différentes de positionnement. On reporte figure 3.17 la moyenne du pourcentage des segments correctement identifiés par les deux méthodes pour 100 réalisations de la trajectoire, pour chaque valeur de la précision du récepteur GPS.

On constate que l'algorithme proposé maintient un pourcentage moyen des segments correctement identifiés autour de 93.6%, tandis que pour l'algorithme topologique, le pourcentage des segments correctement identifiés décroît avec l'augmentation de l'erreur de positionnement GPS. L'algorithme MAP-SIR devient plus performant dès que l'erreur GPS dépasse 7.5m.

Dans une seconde expérimentation, on compare les deux algorithmes lorsque le masquage est partiel. Dans notre implémentation, on utilise les signaux GPS lorsqu'ils sont disponibles, et les filtres MAP-TOP et MAP-SIR lorsqu'ils sont masqués.

Pour évaluer les performances des deux algorithmes, on masque n points GPS consécutifs à un point de départ choisi aléatoirement sur la trajectoire. Pour chaque valeur de n , on génère 100 réalisations de l'expérience avec un point de départ choisi aléatoirement pour chaque réalisation. On reporte dans le tableau 3.4 le pourcentage de segments correctement identifiés en

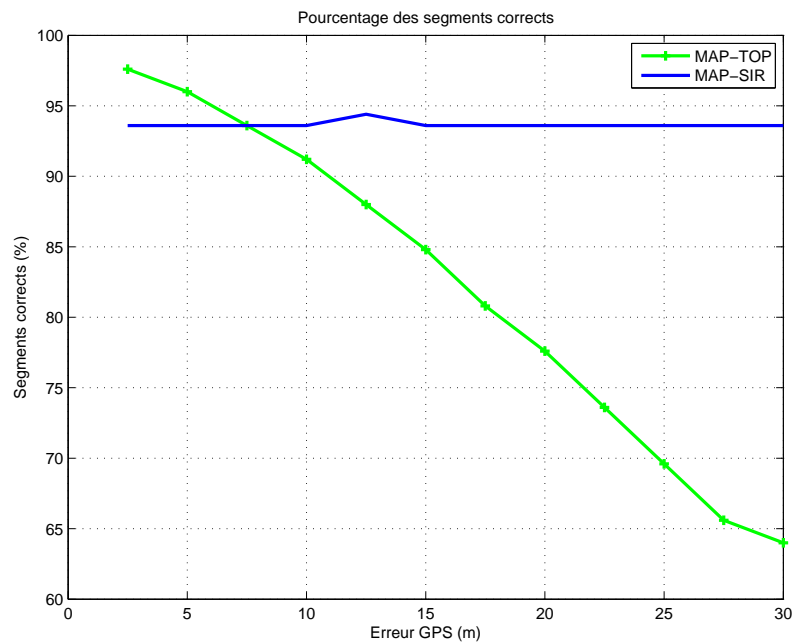


FIGURE 3.17 – Pourcentage des segments correctement identifiés au sujet de différentes précisions GPS pour MAP-TOP, tous les points GPS étant masqués pour MAP-SIR

fonction du pourcentage de points masqués. On présente sur la figure 3.18 l'évolution de cette erreur en fonction du pourcentage de points masqués. La variance de l'erreur GPS (variance de la distance entre la position GPS et la position réelle) est de $0m$ et $10m$.

Points masqués	8%	24%	40%	56%	72%
MAP-TOP	91.2%	91.2%	90.4%	89.6%	88.8%
MAP-SIR	91.2%	91.5%	91.2%	91.5%	92.6%

TABLE 3.4 – Pourcentage de segments corrects en fonction du pourcentage de points masqués (Erreur GPS = $10m$)

On peut conclure que l'approche proposée, qui consiste à utiliser l'algorithme MAP-TOP lorsque les points GPS sont disponibles et MAP-SIR en absence de points GPS, est plus performante que l'approche classique MAP-TOP. En effet, l'algorithme MAP-SIR offre une meilleure performance dans le choix du segment correct par rapport à l'algorithme MAP-TOP, lorsque le nombre des points masqués augmente. On remarque aussi que lorsqu'il existe des masquages partiels, l'algorithme MAP-SIR est plus performant que l'algorithme MAP-TOP pour un récepteur GPS ayant une erreur de $10m$.

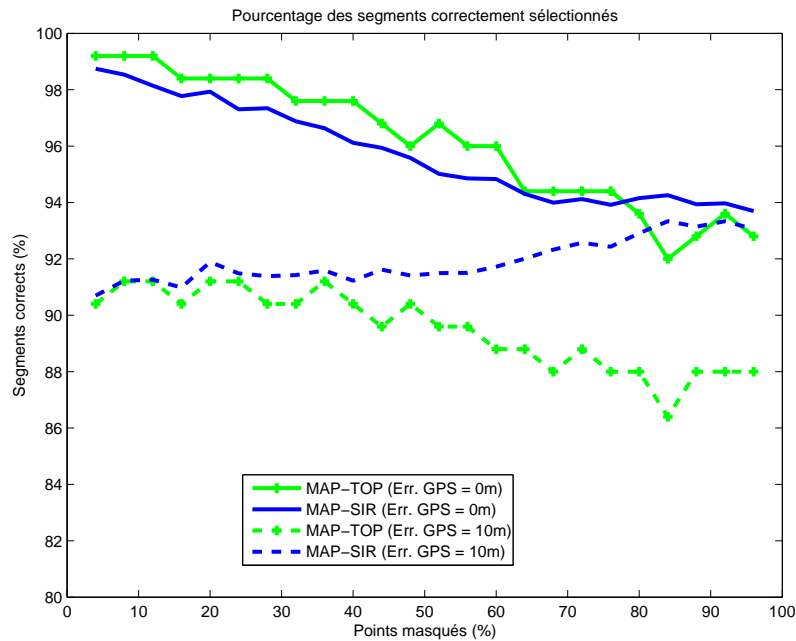


FIGURE 3.18 – Pourcentage des segments correctement identifiés en fonction du pourcentage de points masqués

3.6 Conclusion

Dans ce chapitre, nous avons proposé une nouvelle méthode de Map matching reposant sur le filtre particulière SIR, avec un préfiltrage sur la direction du véhicule. La problématique étudiée est liée à la dégradation de la précision des algorithmes de Map matching classiques en l'absence de signaux GPS. Dans ce cas, la navigation par l'estime est utilisée en se servant des capteurs proprioceptifs. La position est ainsi calculée à partir de la position précédente, d'où une diminution de la précision en fonction du temps.

L'algorithme proposé estime la position du véhicule à partir des observations de vitesse et de direction du véhicule. Deux algorithmes de préfiltrage ont été évalués pour estimer la direction : le filtre circulaire et le filtre particulière circulaire. Deux trajectoires synthétiques ont été utilisées dans les expérimentations pour évaluer la méthode de Map matching : une trajectoire avec une bifurcation en Y et une trajectoire définie dans un contexte réel avec plusieurs virages.

L'expérimentation faite sur les deux algorithmes circulaires de préfiltrage a montré une meilleure performance de ces filtres par rapport au filtre de Kalman qui est sensible aux transitions de direction entre $-\pi$ et π . Les deux algorithmes proposés ont fourni des performances similaires. Nous avons choisi le filtre circulaire pour la suite des expérimentations car il est

moins onéreux en termes de charge de calcul comparé au filtre circulaire particulaire.

Dans l'expérimentation, on montre que le Map matching particulaire fournit de meilleurs résultats lorsqu'il est comparé à un algorithme de Map matching topologique classique. En effet, l'erreur sur la position estimée est plus faible et l'identification du changement de route aux bifurcations est plus rapide. Enfin la méthode proposée est plus robuste au bruit, car sa supériorité reste vraie lorsque l'angle de la bifurcation diminue.

L'apport en précision de l'algorithme de Map matching particulaire proposé est lié au fait que les particules sont re-concentrées sur la bonne route à l'instant de la bifurcation. De ce fait, les bifurcations agissent comme des points clés qui permettent d'éliminer les particules sur les segments improbables et de les re-concentrer sur les segments les plus probables. Cela n'est pas le cas pour l'algorithme topologique car l'erreur sur la position estimée continue d'augmenter puisque l'algorithme repose uniquement sur la navigation à l'estime.

L'expérimentation dans le contexte réel a permis de comparer les algorithmes de Map matching particulaire et topologique en situation de masquage temporaire des signaux GPS dans un milieu urbain. Cette expérimentation a montré que le pourcentage des segments correctement identifiés par le Map matching particulaire est supérieur à celui de l'algorithme topologique lorsque le nombre de points masqués augmente, pour une erreur du récepteur GPS qui dépasse 7.5m.

La performance constatée dans le contexte réel est expliquée d'une part par le gain en précision par la re-concentration des particules à chaque virage comme expliqué précédemment, et d'autre part par l'utilisation d'un processus de décision "souple". En effet, à chaque bifurcation, l'algorithme place des particules sur toutes les routes possibles et évalue leur poids pendant plusieurs itérations. Ceci permet de se repositionner sur la bonne route lorsqu'une erreur de décision a été faite à la bifurcation. En revanche, l'algorithme topologique utilise un processus de décision "dure", sans effet rétroactif, ce qui rend impossible le repositionnement sur la bonne route, à la suite à d'une erreur de décision.

Chapitre 4

L'IMM circulaire

4.1 Introduction

Les algorithmes multi-modèles exécutent en parallèle un ensemble de filtres statistiques associés, à un instant donné, aux dynamiques possibles du système. L'évolution du système est représentée par un ensemble d'équations d'état qui correspondent à ces différentes dynamiques. Les algorithmes multi-modèles estiment l'état et la matrice de covariance par un mixage, selon un modèle markovien, des estimées fournies par chaque filtre. Plusieurs algorithmes ont été discutés en détail dans le chapitre 2, chacun d'eux effectue cette combinaison d'estimées différemment. Le plus répandu de ces algorithmes est l'algorithme multi-modèles interactif IMM reposant sur le filtre de Kalman.

L'une des principales applications de l'IMM concerne la poursuite de cibles ou *tracking*. Par exemple, dans [36], l'algorithme IMM est appliqué dans la poursuite d'un véhicule avec une caméra vidéo. Dans [87], l'IMM est mis en œuvre pour détecter la déviation du véhicule de la route. Dans ce cas, les mesures sont fournies par un LIDAR, un RADAR et une caméra. Le filtre IMM est utilisé dans [72] pour définir la dynamique d'une cible manœuvrante. Une étude comparative des différents filtres IMM peut être trouvée dans [68].

Pour les différentes applications citées précédemment, les mouvements sont supposés linéaires et les bruits gaussiens. Dans ce cas, le filtre IMM est le meilleur compromis entre la précision et le coût de calcul. Cependant, lorsque les modèles ne sont plus linéaires, l'algorithme IMM est moins efficace et peut diverger rapidement [7]. Plusieurs variantes de l'IMM ont été proposées dans les publications scientifiques sur les modèles non linéaires. Dans la plupart des cas, ces algorithmes font appel au filtre de Kalman étendu EKF [2]. Cependant, bien

que l'EKF soit applicable théoriquement sur les systèmes non linéaires, il présente deux défauts en pratique [34, 73]. D'un côté, ce filtre est conçu pour une linéarisation locale du problème et devient instable lors de larges variations de l'état. D'un autre côté, la matrice jacobienne n'est pas triviale pour toutes les applications et son usage entraîne souvent une augmentation de la complexité du calcul.

Dans notre approche, nous considérons un mobile pouvant effectuer un déplacement linéaire selon une direction fixe, une rotation à droite ou une rotation à gauche. Les rotations peuvent avoir des vitesses angulaires différentes. Les modèles considérés sont donc circulaires, la grandeur estimée étant la direction du véhicule. Lorsque l'état est un angle, on utilise souvent le filtre de Kalman étendu. L'approximation gaussienne est valide pour de faibles variations d'angle, car dans ce cas elle reste une bonne approximation des distributions normales circulaires. Mais, cette approximation n'est plus valide pour de larges angles et aussi pour les transitions angulaires entre 0 et 2π . Ces transitions naturelles dans le domaine circulaire sont vues dans le domaine linéaire comme des changements abrupts d'angle. L'utilisation d'une distribution statistique circulaire pour modéliser le problème permet d'apporter des solutions indépendantes de la nature périodique des angles. La distribution statistique de von Mises dans le domaine circulaire constitue l'équivalent de la distribution gaussienne dans le domaine linéaire. Dans ce chapitre, nous allons proposer un algorithme de filtrage multi-modèles interactif circulaire qu'on notera dans la suite CIMM. C'est un algorithme défini dans le cadre bayésien et qui utilise la distribution circulaire de von Mises. Cet algorithme repose également sur l'opérateur somme pondérée circulaire et sur le filtre circulaire récursif, proposés dans [6, 80, 77].

L'algorithme CIMM est utilisé pour le Map matching. Comme discuté au chapitre 1, l'une des tâches fondamentales de Map matching consiste à sélectionner correctement le segment sur lequel le véhicule se déplace dans un réseau de routes. Dans une approche récursive, cette tâche est subdivisée en deux sous-tâches séquentielles : détecter un changement de segment, puis identifier le nouveau segment dans la base de données de la carte numérique. La plupart des systèmes de navigation multi-capteurs modernes sont composés d'un capteur de direction (magnétomètre/gyroscope) et d'un capteur de position (GNSS). Ces deux sous-tâches sont réalisées séparément en se référant aux observations issues des deux capteurs, puis elles sont fusionnées pour aboutir à la décision finale. Dans ce chapitre, nous proposons d'utiliser le filtre CIMM pour détecter le changement et sélectionner le segment dans la base de données à partir des observations de la direction.

Ce chapitre est organisé comme suit : après cette introduction, nous allons poser dans une

seconde partie le problème de l'estimation multi-modèles et de la modélisation circulaire. La troisième partie est consacrée à la présentation du filtre multi-modèles circulaire proposé. Son application au Map matching est présentée dans la quatrième partie. Enfin c'est dans la cinquième partie que le filtre proposé est évalué sur données synthétiques et réelles. Ce chapitre sera terminé par une conclusion.

4.2 Position du problème

4.2.1 Limites de l'approche particulière circulaire

Nous avons présenté au chapitre 3 l'algorithme particulière de Map matching avec un pré-filtrage sur la direction du véhicule et nous avons démontré les performances de cet algorithme sur des données synthétiques, en le comparant à un algorithme de Map matching topologique. Toutefois, l'étude réalisée a montré aussi des limitations que nous discutons dans cette section.

Le pré-filtrage de la direction du véhicule repose sur un filtre circulaire défini dans le cadre bayésien et utilisant la distribution de von Mises avec un schéma de prédiction/correction similaire au filtre de Kalman. Le modèle sur lequel repose ce filtre est décrit par les équations d'état (3.28) et (3.29). Dans l'équation (3.29), on suppose la variation angulaire quasi-constante. Toute manœuvre de rotation est considérée comme faisant partie du bruit d'état noté \tilde{v}_{k-1} de facteur de concentration $\kappa_{\tilde{Q}}$. Ici on se trouve devant un dilemme dans le réglage du paramètre $\kappa_{\tilde{Q}}$: diminuer $\kappa_{\tilde{Q}}$ permet d'assurer une meilleure réactivité du filtre aux variations angulaires mais au détriment d'un filtrage médiocre des bruits de mesure. Augmenter $\kappa_{\tilde{Q}}$, en revanche, assure un meilleur filtrage du bruit, mais oblige le filtre à réagir moins rapidement aux variations angulaires, d'où un retard dans la prise en considération de la manœuvre. Ce retard est constaté dans les résultats expérimentaux du chapitre 3 par une augmentation de l'erreur de positionnement à l'instant des bifurcations. L'algorithme de Map matching particulière en est directement influencé. En effet, la manœuvre étant détectée en retard, les particules placées sur le segment où la manœuvre a lieu gardent toujours un poids plus faible, d'où une détérioration de la précision à l'instant de la bifurcation.

Nous sommes donc devant un problème d'estimation adaptative [8] où les manœuvres de l'objet observé imposent des variations de comportement auxquelles le modèle mathématique doit s'adapter. Plusieurs méthodes ont été exposées dans [8] pour résoudre ce problème :

- **Méthode de l'ajustement continu du niveau du bruit du processus** (*continuous noise*)

level adjustment). La cible est suivie par un filtre auquel un faible niveau de bruit de modèle est considéré. Le niveau de bruit est représenté par sa variance ou sa matrice de covariance. Une manœuvre se manifeste par une “grande” innovation. Cette méthode consiste alors à implémenter une procédure de détection pour de telles innovations de façon à ce que lorsqu’elles dépassent un seuil prédéfini, on augmente la variance du bruit du processus pour permettre au filtre de s’y adapter plus rapidement. Une fois la variation de l’innovation en dessous d’un second seuil, la variance du bruit du processus est rétablie à sa valeur normale. Il n’existe pas de manière exacte pour le choix des deux seuils de détection. Ils sont souvent déterminés à partir de données expérimentales (réalisations de Monte Carlo). Les innovations sont surveillées à travers une moyenne mobile d’observations définies sur une fenêtre glissante.

- **Méthode de l’estimation de l’entrée de commande** (IE : *Input Estimation*) où l’on estime un paramètre supplémentaire qui est l’entrée de commande inconnue en observant les innovations à travers une fenêtre glissante. Un test statistique effectué sur l’estimation de la commande permet de décider si l’on est en présence d’une manœuvre, et dans ce cas l’estimation de l’état est corrigée par un terme supplémentaire qui prend en compte l’entrée de commande estimée.
- **Méthode de filtrage à dimension variable** (VSD : *Variable Dimension Filtering*) où l’on ajoute temporairement de nouvelles variables au vecteur d’état au moment de la manœuvre pour mieux représenter le modèle réel à cet instant. On utilise ainsi deux modèles : un modèle simple représentant l’état “calme” du système et un modèle d’ordre plus élevé représentant la manœuvre. L’objectif est d’améliorer les performances de la poursuite en limitant l’utilisant du modèle complexe à la manœuvre. En effet, dans le cas d’un mouvement à vitesse constante, l’ajout par exemple d’une composante d’accélération augmente inutilement, les erreurs sur la position et la vitesse, ainsi que la charge du calcul.

Les performances de ces méthodes ont été évaluées dans [8]. Une étude expérimentale a montré que la méthode d’ajustement du bruit et la méthode VSD fournissent des résultats comparables et sont plus performantes et moins onéreuses en charge de calcul que la méthode IE. Cependant, il a été indiqué dans [8] qu’aucune de ces trois méthodes n’a pu abaisser l’erreur sur l’estimation de la position pendant la manœuvre au dessous de l’erreur de mesures, ce qui permet de conclure que l’adaptation des algorithmes n’est pas assez rapide. Il a été montré dans [8] que l’approche multi-modèles apporte une solution à ce problème en faisant appel à plusieurs modèles simples, chacun adapté aux différentes manœuvres possibles de l’objet observé. Le chapitre 2 a été entièrement consacré à l’étude des différents aspects de cette approche. Dans ce

chapitre nous proposons d'utiliser l'approche multi-modèles pour améliorer les performances de l'algorithme de Map matching particulière.

Un autre problème rencontré avec le Map matching particulière concerne le placement des particules arrivées à la fin d'une route. En effet, la méthode décrite au chapitre 3 repose sur un échantillonnage aléatoire uniforme sur toutes les routes connectées à la route actuelle. Ainsi, les particules sont distribuées de la même façon sur toutes les routes, même sur celles qui ont très peu de probabilité d'être choisies si l'on considère la nature de la manœuvre du véhicule. On introduit dans ce cas un retard de plusieurs pas de calcul, avant que l'algorithme particulière re-concentre ses particules sur la route correcte, produisant ainsi une erreur de positionnement au moment de la bifurcation.

Nous proposons de positionner les particules arrivées à la bifurcation en prenant en compte la nature de la manœuvre du véhicule et en donnant ainsi plus d'importance aux routes qui se trouvent dans la direction de cette manœuvre. Cette méthode s'appuie sur le filtrage multi-modèles et sera présentée dans la suite de ce chapitre.

4.2.2 L'approche proposée

L'approche multi-modèles est appliquée au Map matching particulière pour améliorer la qualité du Map matching sur plusieurs aspects. L'idée de base du filtrage multi-modèles est d'exécuter en parallèle un ensemble de filtres reposant sur des modèles simples, chacun correspondant à un comportement du système étudié.

Dans le cas du Map matching particulière, les manœuvres qui nous intéressent sont les virages à gauche ou à droite, avec des vitesses de rotation différentes. Comme expliqué au chapitre 3, les virages fonctionnent comme des amers qui entraînent la re-concentration des particules sur les bifurcations et diminuent ainsi l'erreur sur la position estimée. Dans le cas d'un mouvement sans variation angulaire sur une même route, l'erreur de position continue à augmenter à cause du cumul de l'erreur sur la vitesse.

Le filtre multi-modèles que nous avons retenu est le filtre multi-modèles interactif ou IMM. Comme expliqué au chapitre 2, c'est un filtre sous-optimal qui fait partie des filtres multi-modèles coopératifs où les différents modèles fonctionnent en interaction pour fournir la meilleure estimée. Contrairement aux méthodes de détection de manœuvres mentionnées dans la section

4.2.1, le filtre IMM ne s'appuie pas sur un test d'hypothèse pour déterminer le modèle à appliquer. Son algorithme calcule, à chaque itération, la probabilité de tous les modèles. L'estimée finale est une somme des estimées issues de chaque modèle, pondérée par sa probabilité. L'interaction entre les filtres élémentaires permet d'éviter la divergence des filtres associés aux modèles non applicables (et donc moins probable) en minimisant leur influence par un mixage pondéré (les poids étant définis à partir des probabilités de chaque modèle) des estimées de tous les filtres. En plus de son utilisation dans le calcul de l'estimée finale, la probabilité de modèle nous informe aussi du modèle le plus proche du comportement observé et peut ainsi servir de base pour identifier la nature de la manœuvre du véhicule.

La nouveauté apportée est l'utilisation de l'approche circulaire dans le filtre IMM qui sera appelé dans la suite filtre IMM circulaire ou CIMM. Ceci est expliqué par la nature de la grandeur estimée qui est la direction du véhicule. Comme démontré dans [78, 26], les filtres circulaires sont plus adaptés à l'estimation des angles grâce à l'utilisation de la distribution de von Mises qui est l'équivalent de la distribution normale dans le domaine circulaire.

Le filtre CIMM apportera donc deux améliorations au Map matching particulière : d'une part il fournira une meilleure estimation de la direction du véhicule en améliorant la réponse du filtre à la manœuvre (prise en compte rapide du début et de la fin de la manœuvre), et d'autre part, il fournira, à travers les probabilités de modèle, une information sur la nature de la manœuvre. Cette information permettra de placer un maximum de particules sur les segments situés dans le sens de la manœuvre observée.

4.2.3 Les modèles circulaires

En général, toutes les méthodes de poursuite de cible font appel à des modèles mathématiques précis et sont représentés souvent sous forme d'un modèle d'état construit avec une équation qui modélise la dynamique du système et une équation qui modélise les observations [50] :

$$x_{k+1} = F_k x_k + G_k^u u_k + G_k w_k \quad (4.1)$$

$$z_k = H_k x_k + v_k \quad (4.2)$$

Souvent, l'entrée de commande u_k est inconnue, ainsi que la matrice de transition F_k et les propriétés statistiques du bruit w_k .

Dans la plupart des méthodes de poursuite de cible, on considère deux catégories de modèles : modèle avec manœuvres et modèle sans manœuvre. Ce dernier correspond généralement

à un mouvement uniforme à vitesse constante. Tout autre type de mouvement est classé dans la première catégorie.

Une étude détaillée des différents types de modèles utilisés dans les applications de poursuite a été présentée dans [50]. Si l'on considère un vecteur d'état $x = [\xi, \dot{\xi}, \eta, \dot{\eta}]$ représentant la position et la vitesse d'un véhicule en mouvement sur deux dimensions, un modèle sans manœuvre peut s'écrire à partir de l'équation (4.1) :

$$x_{k+1} = F^{CV} x_k + G_k w_k \quad (4.3)$$

avec :

$$F^{CV} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} ; G_k = \begin{bmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & T^2/2 \\ 0 & T \end{bmatrix} \quad (4.4)$$

T étant le pas de calcul. Ici on considère l'entrée de commande u_k nulle et la vitesse quasi-constante (CV : *Constant Velocity*). Ce modèle peut accepter de faibles accélérations qui sont modélisées par la composante du bruit $G_k w_k$.

Pour modéliser les virages, la variation angulaire de la direction du véhicule ω est ajoutée au vecteur d'état $x = [\xi, \dot{\xi}, \eta, \dot{\eta}, \omega]$, le modèle du mouvement avec manœuvres de rotation s'écrit alors [40] :

$$x_{k+1} = F_k^{CST} x_k + G_k w_k \quad (4.5)$$

avec :

$$F_k^{CST} = \begin{bmatrix} 1 & \frac{\sin \omega(k-1)T}{\omega(k-1)} & 0 & \frac{1-\cos \omega(k-1)T}{\omega(k-1)} & 0 \\ 0 & \cos \omega(k-1)T & 0 & -\sin \omega(k-1)T & 0 \\ 0 & \frac{1-\cos \omega(k-1)T}{\omega(k-1)} & 1 & \frac{\sin \omega(k-1)T}{\omega(k-1)} & 0 \\ 0 & \sin \omega(k-1)T & 0 & \cos \omega(k-1)T & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} ; G_k = \begin{bmatrix} T^2/2 & 0 & 0 \\ T & 0 & 0 \\ 0 & T^2/2 & 0 \\ 0 & T & 0 \\ 0 & 0 & T \end{bmatrix} \quad (4.6)$$

Le système sans manœuvre, modélisé par l'équation (4.3), peut être implémenté avec un filtre de Kalman classique. En revanche, le système avec une manœuvre de rotation, décrit par l'équation (4.5), est non linéaire. Son implémentation fait appel à d'autres filtres plus coûteux en complexité de calcul comme l'EKF [9, 30] ou l'UKF [40].

Si on considère que le virage se fait avec un taux de rotation constant ω , on peut réduire de nouveau le vecteur d'état à 4 variables : $x = [\xi, \dot{\xi}, \eta, \dot{\eta}]$, et le système devient linéaire avec une matrice de transition :

$$F^{CT} = \begin{bmatrix} 1 & \frac{\sin \omega T}{\omega} & 0 & \frac{1 - \cos \omega T}{\omega} \\ 0 & \cos \omega T & 0 & -\sin \omega T \\ 0 & \frac{1 - \cos \omega T}{\omega} & 1 & \frac{\sin \omega T}{\omega} \\ 0 & \sin \omega T & 0 & \cos \omega T \end{bmatrix} \quad (4.7)$$

Ce mouvement est connu sous le nom de “Virage coordonnée” (CT : *Coordinated Turn*) [8, 50, 30] où la cible se déplace à vitesse constante avec un taux de rotation constant. Cependant en pratique, cette simplification n'est pas valable car le taux de rotation ω n'est pas constant pour tous les virages. Par conséquent, et afin d'éviter l'utilisation du modèle non linéaire de l'équation (4.5), on peut considérer plusieurs modèles CT avec des taux de rotations différents sous une implémentation multi-modèles. Cette méthode a été mise en œuvre dans [68] où 9 modèles ont été utilisés dont un modèle sans manœuvre CV, quatre modèles CT et quatre autres modèles avec accélération tangentielle constante.

Dans le cas de notre implémentation, la position du véhicule (x, y) est estimée par le Map matching particulière étudié au chapitre 3. Nous désirons utiliser le filtre IMM pour améliorer l'estimation de la direction du véhicule pendant les manœuvres, et en même temps, utiliser les probabilités de modèles issues de l'IMM pour améliorer la distribution des particules sur les virages pouvant exister à droite ou à gauche.

Nous nous intéressons uniquement à l'angle de direction. Les modèles que nous considérons sont donc :

- *Mouvement sans manœuvre* : où l'angle de direction est quasi-constant et la vitesse angulaire nulle,
- *Mouvement avec manœuvres de rotation* à gauche ou à droite pour modéliser les virages, où le taux de rotation est considéré quasi-constant.

Le mouvement sans manœuvre est modélisé par :

$$\theta_k = \theta_{k-1} + \nu_{k-1} \quad (4.8)$$

θ_k et θ_{k-1} sont les directions du véhicule aux instants k et $k - 1$ respectivement. ν_{k-1} est le bruit du modèle sur θ_k . Ce bruit additif est distribué selon une distribution de von Mises centrée sur zéro et ayant un paramètre de concentration de κ_Q . Ce bruit permet au modèle de s'adapter aux faibles variations angulaires nécessaires pour ajuster l'état aux mesures.

Le mouvement avec manœuvres de rotation sur un virage est modélisé par :

$$\theta_k = \theta_{k-1} + \xi + \nu_{k-1} \quad (4.9)$$

Dans cette équation, définie selon le modèle d'état (4.1), le taux de rotation est supposé constant et défini par la commande $|\xi|$ pour une rotation à droite et $-|\xi|$ pour une rotation à gauche. Il est ainsi possible de créer plusieurs modèles avec des taux de rotation différents pour modéliser les manœuvres aux virages.

4.3 L'IMM circulaire

4.3.1 L'estimation circulaire

L'estimateur circulaire

Considérons un ensemble d'observations bruitées d'un angle θ suivant la distribution circulaire de von Mises. Afin d'estimer l'angle θ , il faudrait estimer les paramètres statistiques de la distribution de von Mises au sens du maximum de vraisemblance. La distribution de von Mises est définie par sa moyenne et son paramètre de concentration.

La figure 4.1 montre clairement que lorsque la moyenne n'est pas centrée sur l'intervalle de mesure considéré, dans ce cas $[0, 2\pi[$, la distribution n'a pas qu'un seul maximum. Cela nous oblige de recourir aux outils statistiques définis dans le domaine circulaire plutôt que ceux du domaine linéaire.

Estimation de la moyenne

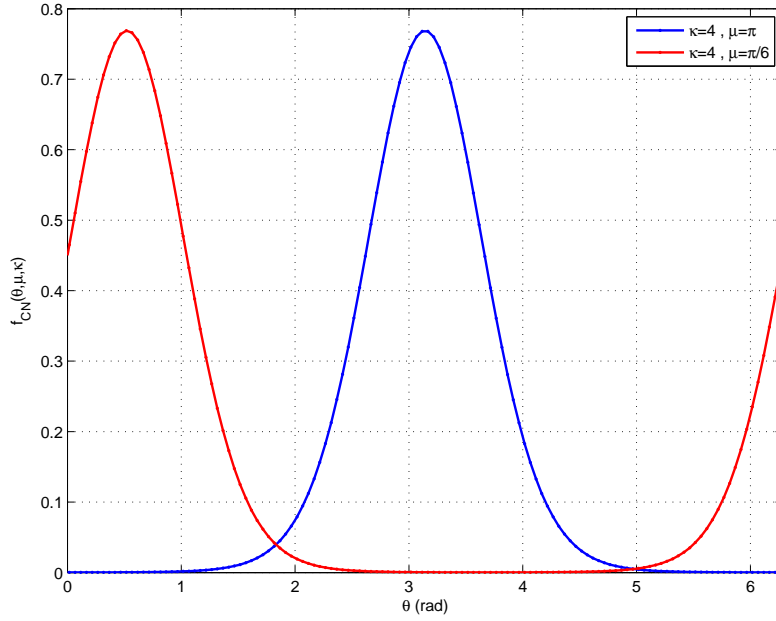


FIGURE 4.1 – Distribution de von Mises

Dans le domaine circulaire, la moyenne de plusieurs angles est obtenue intuitivement par la représentation complexe des observations qui prend en considération la nature périodique des angles. Ainsi, la moyenne s'exprime par :

$$\hat{\mu} = \arg \left(\sum_{k=1}^n e^{j\phi_k} \right) \quad (4.10)$$

où j est le nombre complexe tel que $j^2 = -1$. L'estimateur du maximum de vraisemblance de la moyenne de plusieurs angles suivant la distribution de von Mises est défini dans [33] par (4.11) . Il coïncide avec la moyenne (4.10).

$$\hat{\mu} = \arctan^* \left(\sum_{k=1}^n \sin(\phi_k), \sum_{k=1}^n \cos(\phi_k) \right) \quad (4.11)$$

\arctan^* est la réciproque de la tangente qui dépend du quadrant. Elle est définie dans la section 3.4.2.

Estimation du paramètre de concentration

Pour n observations suivant une distribution de von Mises de moyenne μ et de paramètre de concentration κ , l'estimateur optimal au sens du maximum de vraisemblance du paramètre de

concentration $\hat{\kappa}$ est défini par [33] :

$$\hat{\kappa} = A^{-1} \left(\frac{1}{n} \sum_{k=1}^n \cos(\phi_k) - \hat{\mu} \right) \quad (4.12)$$

$$= A^{-1} \left(\frac{1}{n} \left| \sum_{k=1}^n e^{j\phi_k} \right| \right) \quad (4.13)$$

La fonction $A^{-1}(\cdot)$ est définie dans la section 3.4.2, son calcul est décrit dans [76, 27].

La fusion circulaire

Considérons N mesures circulaires $\phi_k^{(1)}, \dots, \phi_k^{(N)}$ que l'on souhaite fusionner en une seule mesure circulaire $\phi_{f,k}$. On suppose que toutes les mesures sont indépendantes et suivent une distribution de von Mises de moyenne θ_k et de paramètres de concentration respectifs $\kappa_R^{(1)}, \dots, \kappa_R^{(N)}$. L'opérateur de fusion de variables circulaires distribuées selon la distribution de von Mises est défini au sens du maximum de vraisemblance [6, 76]. Cet opérateur de fusion intègre la périodicité des variables circulaires et préserve l'indépendance de la direction zéro de référence. La fonction de vraisemblance de l'état θ_k sachant les mesures est ainsi définie par :

$$\Lambda(\phi_k^{(1)}, \dots, \phi_k^{(N)} | \theta_k) = f_{CN}(\phi_k^{(1)} | \theta_k, \kappa_R^{(1)}) f_{CN}(\phi_k^{(2)} | \theta_k, \kappa_R^{(2)}) \dots f_{CN}(\phi_k^{(N)} | \theta_k, \kappa_R^{(N)}) \quad (4.14)$$

avec :

$$f_{CN}(\phi_k^{(i)} | \theta_k, \kappa_R^{(i)}) = \frac{1}{2\pi I_0 \kappa_R^{(i)}} \exp \left(\kappa_R^{(i)} \cos(\phi_k^{(i)} - \theta_k) \right) \quad (4.15)$$

D'où :

$$\Lambda(\phi_k^{(1)}, \dots, \phi_k^{(N)} | \theta_k) = \frac{1}{2\pi I_0 \kappa_R^{(1)}} \dots \frac{1}{2\pi I_0 \kappa_R^{(N)}} \exp \left(\sum_{i=1}^N \kappa_R^{(i)} \cos(\phi_k^{(i)} - \theta_k) \right) \quad (4.16)$$

On calcule ensuite le logarithme de la fonction de vraisemblance :

$$l = \ln \left(\Lambda(\phi_k^{(1)}, \dots, \phi_k^{(N)} | \theta_k) \right) = - \sum_{i=1}^N \ln \left(2\pi I_0 \kappa_R^{(i)} \right) + \sum_{i=1}^N \kappa_R^{(i)} \cos(\phi_k^{(i)} - \theta_k) \quad (4.17)$$

Le maximum de vraisemblance noté $\phi_{f,k}$ correspond à la valeur de θ_k qui annule la dérivée $\frac{\partial l}{\partial \theta_k}$ tout en ayant une dérivée seconde positive : $\frac{\partial^2 l}{\partial^2 \theta_k} > 0$. On démontre dans [76] que ce

maximum est obtenu pour :

$$\phi_{f,k} = \arctan^*(S_k, C_k) = \arg \left(\sum_{i=1}^N e^{j\kappa_R^{(i)} \phi_k^{(i)}} \right) \quad (4.18)$$

$$(4.19)$$

avec :

$$S_k = \sum_{i=1}^N \kappa_R^{(i)} \sin(\phi_k^{(i)}) \quad (4.20)$$

$$C_k = \sum_{i=1}^N \kappa_R^{(i)} \cos(\phi_k^{(i)}) \quad (4.21)$$

Le paramètre de concentration du résultat de la fusion $\kappa_{f,k}$ est obtenu par [76] :

$$\kappa_{f,k} = \sqrt{S_k^2 + C_k^2} \quad (4.22)$$

Une remarque fondamentale que l'on indique ici, note que, à l'opposé de l'opérateur optimal du domaine linéaire (la somme pondérée), le paramètre de concentration de l'opérateur de fusion dépend des mesures. Cependant, on démontre dans [76] que la variance circulaire de la moyenne des mesures fusionnées est toujours inférieure à la variance circulaire de chaque mesure.

4.3.2 Le filtre IMM circulaire

Principe du filtre IMM circulaire

Le filtre IMM est utilisé dans les applications où l'on suppose que le modèle peut changer au cours du temps. Dans ce cas, les différentes dynamiques du mobile sont difficiles à définir en un modèle unique. L'IMM considère, à tout moment, tous les modèles applicables et calcule leur probabilité afin d'estimer l'état du système.

Considérons un système et son modèle défini parmi r modèles $\{M^1, \dots, M^r\}$. On suppose connue la probabilité initiale $\mu_0^j = P\{M_0^j\}$ de chaque modèle M^j . Les probabilités de transition $p_{ij} = P\{M_k^j | M_{k-1}^i\}$ du modèle i à l'instant $k-1$ au modèle j à l'instant k sont supposées aussi connues. Ces transitions caractérisent complètement une chaîne de Markov et définissent un système à commutation markovienne. Le filtrage d'état optimal dans le cas multi-modèles nécessite l'exécution des filtres associés à toutes les transitions possibles. Ainsi, pour r modèles, r^k filtres optimaux doivent être exécutés à l'instant k . Afin de réduire cette charge de calcul, on

a recours à des méthodes sous-optimales décrites dans le chapitre 2.

Le filtre IMM offre une bonne précision de l'estimation avec un coût de calcul optimal [9]. Il se déroule en trois étapes : *interaction/mixage*, *filtrage* et *combinaison*. A chaque instant k , les états précédents sont mixés puis utilisés par les filtres de Kalman associés aux différents modèles pour créer r estimées. Celles-ci sont combinées pour produire une estimée finale unique et sa matrice de covariance. Les poids utilisés dans la combinaison sont définis par les probabilités de chaque modèle. Les détails des calculs théoriques du filtre IMM sont fournis dans la section 2.5.3.

Puisque nous faisons appel dans notre problème à des données angulaires qui sont les mesures de la direction du véhicule, le filtre de Kalman est remplacé par un filtre circulaire récursif défini dans le cadre bayésien avec la distribution de von Mises. L'architecture globale du filtre multi- modèles circulaire CIMM est illustrée sur la figure 4.2.

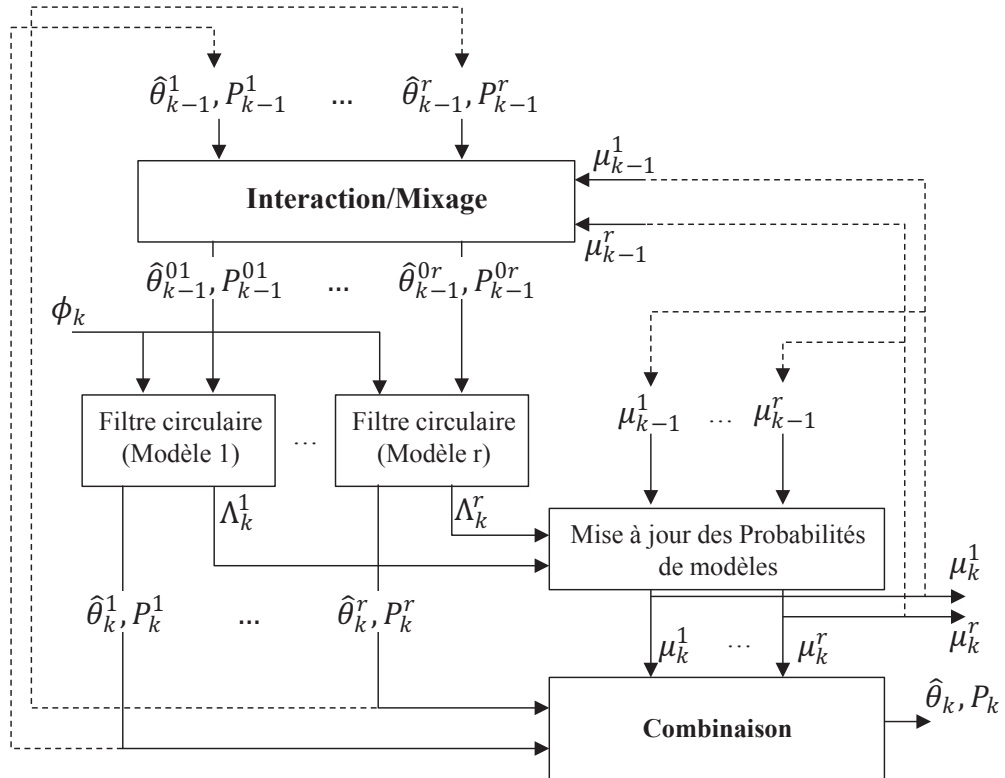


FIGURE 4.2 – Structure du filtre CIMM

Dans la suite du texte, nous allons décrire chaque élément de cette architecture.

Équations d'état et de mesure

Chaque modèle du filtre CIMM est décrit par une équation d'état et une équation de mesure.

Équation d'état

L'angle de direction θ_k définit l'état à l'instant k :

$$\theta_k = \theta_{k-1} + \xi + \nu_{k-1} \quad (4.23)$$

ν_{k-1} est le bruit du modèle sur θ_k . Il représente l'erreur du modèle d'état. Ce bruit additif est distribué selon une distribution de von Mises centrée sur zéro et ayant un paramètre de concentration de κ_Q . Pour $\xi = 0$ on a le mouvement sans manœuvre et pour $\xi \neq 0$ on a les mouvements avec manœuvres.

Équation de mesure

Les données fournies par le magnétomètre constituent les mesures ϕ_k de l'angle θ_k . L'équation de mesure est donnée par :

$$\phi_k = \theta_k + \omega_k \quad (4.24)$$

ω_k est le bruit de mesure. Il est distribué selon une distribution de von Mises centrée avec un paramètre de concentration κ_R .

Étape 1 : Interaction / Mixage

Étape 1a : Traitement des probabilités conditionnelles des modèles

On calcule la probabilité conditionnelle du modèle $\mu^{i|j}$ qui représente la probabilité du modèle M^i à l'instant $k-1$, sachant M^j le modèle à l'instant k [9].

$$\mu_{k-1}^{i|j} = \frac{1}{\bar{c}_j} p_{ij} \mu_{k-1}^i \quad (4.25)$$

$$\bar{c}_j = \sum_{i=1}^r p_{ij} \mu_{k-1}^i \quad (4.26)$$

Avec :

- μ_{k-1}^i : Probabilité du modèle M^i à l'instant $k-1$
- \bar{c}_j : Facteur de normalisation
- p_{ij} : Probabilité de transition du modèle M^i au modèle M^j

Étape 1b : Mixage Dans cette étape, les estimées délivrées par chaque filtre à l'instant $k-1$ sont combinées. Les poids sont définis par les probabilités conditionnelles des modèles $\mu_{k-1}^{i|j}$. On calcule les données initiales pour tous les filtres avec une somme pondérée définie dans le domaine circulaire $[6, 80]$:

$$\hat{\theta}_{k-1}^{0j} = \arctan^* \left(\sum_{i=1}^r \mu_{k-1}^{i|j} \sin(\hat{\theta}_{k-1}^i), \sum_{i=1}^r \mu_{k-1}^{i|j} \cos(\hat{\theta}_{k-1}^i) \right) \quad (4.27)$$

$$P_{\theta_{k-1}}^{0j} = \sqrt{\left(\sum_{i=1}^r \mu_{k-1}^{i|j} \sin(\hat{\theta}_{k-1}^i) \right)^2 + \left(\sum_{i=1}^r \mu_{k-1}^{i|j} \cos(\hat{\theta}_{k-1}^i) \right)^2}$$

Avec $\hat{\theta}_{k-1}^{0j}$, $P_{\theta_{k-1}}^{0j}$ la valeur intermédiaire (après mixage) de l'angle et du paramètre de concentration associés au filtre j .

Étape 2 : Filtrage

Étape 2a : Filtre circulaire associé à chaque modèle

Chaque modèle M^j est associé à un filtre circulaire récursif défini par une étape de prédiction et une étape de correction [80, 7].

Prédiction :

$$\hat{\theta}_{k|k-1}^j = \hat{\theta}_{k-1}^{0j} + \xi_j \quad (4.28)$$

$$P_{\theta_{k|k-1}}^j = A^{-1} \left(A(P_{\theta_{k-1}}^{0j}) A(\kappa_{Q^j}) \right) \quad (4.29)$$

Avec :

- $\hat{\theta}_{k|k-1}^j$, $P_{\theta_{k|k-1}}^j$: Angle et paramètre de concentration prédit pour le modèle M^j
- κ_{Q^j} : Paramètre de concentration du bruit d'état défini pour le modèle M^j

où ξ_j est le paramètre qui définit le type de mouvement circulaire.

Correction :

$$\hat{\theta}_k^j = \arctan^*(S_{\theta_k}^j, C_{\theta_k}^j) \quad (4.30)$$

$$P_{\theta_k}^j = \sqrt{C_{\theta_k}^{j2} + S_{\theta_k}^{j2}} \quad (4.31)$$

Avec :

$$C_{\theta_k}^j = P_{\theta_{k|k-1}}^j \cos(\hat{\theta}_{k|k-1}^j) + \kappa_R \cos(\phi_k) \quad (4.32)$$

$$S_{\theta_k}^j = P_{\theta_{k|k-1}}^j \sin(\hat{\theta}_{k|k-1}^j) + \kappa_R \sin(\phi_k) \quad (4.33)$$

- ϕ_k : Mesure obtenue à l'instant k
- κ_R : Paramètre de concentration du bruit de la mesure d'angle

Étape 2b : Calcul de la vraisemblance pour chaque filtre

Dans notre application, nous mesurons uniquement l'angle. Ainsi, la vraisemblance $v_{\theta_k}^j$ est définie pour chaque modèle j par l'innovation :

$$v_{\theta_k}^j = \phi_k - \hat{\theta}_{k|k-1}^j \quad (4.34)$$

Son paramètre de concentration est donné par :

$$S_{v_{\theta_k}}^j = A^{-1}(A(P_{\theta_{k|k-1}}^j)A(\kappa_R)) \quad (4.35)$$

La fonction de vraisemblance est une distribution de von Mises. Nous calculons la valeur de la vraisemblance comme suit :

$$\Lambda_k^j = \mathcal{CN}(v_{\theta_k}^j | 0, S_{v_{\theta_k}}^j) \quad (4.36)$$

La valeur Λ_k^j est d'autant plus élevée que la mesure ϕ_k est proche de l'estimée $\hat{\theta}_{k|k-1}^j$ fournie par le filtre associé au modèle M^j .

Étape 2c : Calcul des probabilités de modèle

La probabilité de chaque modèle est déterminée à partir de la fonction de vraisemblance

définie dans l'étape 2b.

$$\mu_k^j = \frac{1}{c} \cdot \Lambda_k^j \bar{c}_j \quad (4.37)$$

Avec :

- \bar{c}_j : Donné par l'équation (4.26)
- c : Facteur de normalisation défini par :

$$c = \sum_{j=1}^r \Lambda_k^j \bar{c}_j \quad (4.38)$$

Étape 3 : Combinaison

Les estimées fournies par chaque filtre sont combinées en utilisant une somme pondérée définie dans le domaine circulaire. Les poids sont les probabilités des modèles. Ceci permet d'augmenter la contribution du modèle le plus probable dans le calcul de l'estimée finale. Cette estimée finale est obtenue par :

$$\hat{\theta}_k = \arctan^* \left(\sum_{j=1}^r \mu_k^j \sin(\hat{\theta}_k^j), \sum_{j=1}^r \mu_k^j \cos(\hat{\theta}_k^j) \right) \quad (4.39)$$

Le paramètre de concentration de l'estimée finale est donné par :

$$P_{\theta_k} = \sqrt{\left(\sum_{j=1}^r \mu_k^j \sin(\hat{\theta}_k^j) \right)^2 + \left(\sum_{j=1}^r \mu_k^j \cos(\hat{\theta}_k^j) \right)^2} \quad (4.40)$$

Nous allons décrire dans la suite l'intégration du filtre CIMM à l'algorithme de map matching particulière.

4.4 Map matching circulaire particulière avec pré-filtrage IMM

4.4.1 Architecture du système avec pré-filtrage IMM

On a montré au chapitre 3 que le prétraitement des observations de la direction du véhicule par un filtre circulaire améliore la précision de l'estimation. Cependant, la courbe de la dispersion circulaire figure 3.12 montre un pic d'erreur au moment du virage. Ce phénomène est expliqué par le changement du comportement du véhicule et par le retard de réaction du filtre circulaire par rapport à ce changement.

On désire montrer l'avantage du filtre CIMM dans l'amélioration de la précision de l'estimation de la direction au virage par l'adoption du modèle adéquat. Nous proposons d'estimer la direction du véhicule qui peut se déplacer en mouvement rectiligne ou faire un virage à droite ou à gauche. Pendant la rotation du véhicule, le taux de rotation (variation d'angle) est supposé constant et défini par $|\xi_j|$ en [rd] pour une rotation à droite et $-|\xi_j|$ pour une rotation à gauche. On représente figure 4.3 la nouvelle architecture du système de map matching intégrant le filtre CIMM.

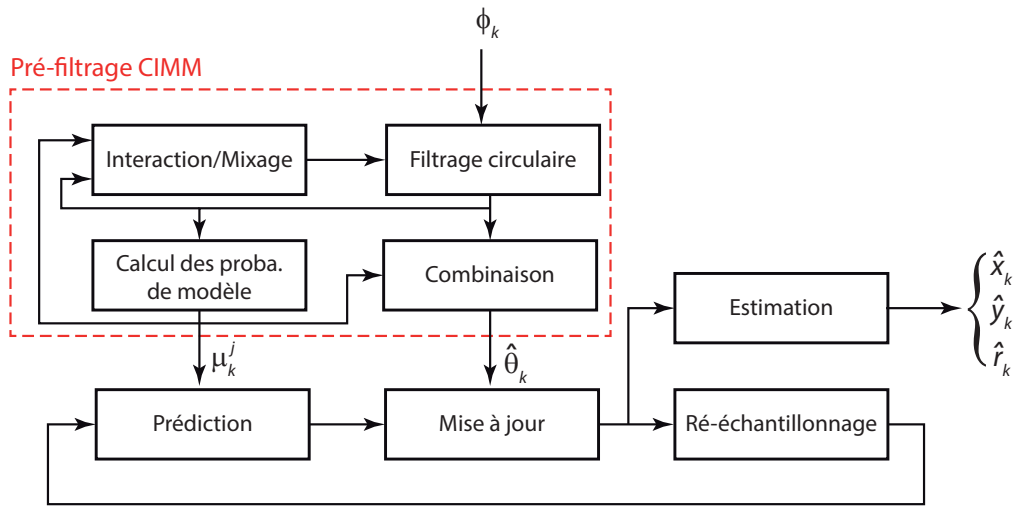


FIGURE 4.3 – Architecture du filtre de Map matching avec pré-filtrage CIMM

Dans le filtre CIMM et pour l'expérimentation de la bifurcation en Y, on utilise les deux modèles suivants :

- **Modèle 1** : Mouvement en ligne droite avec angle quasi-constant
- **Modèle 2** : Mouvement avec manœuvres de rotation à gauche

Dans l'expérimentation du contexte réel, les virages sont présents dans les deux sens ; on fait alors appel à un troisième modèle :

- **Modèle 3** : Mouvement avec manœuvres de rotation à droite

Tous ces modèles sont décrits par l'équation (4.28) avec un taux de rotation $\xi_1 = 0$ pour le modèle 1, un taux $\xi_2 = 0.15 \text{ rd}$ pour la bifurcation en Y, et des taux $\xi_2 = 0.4 \text{ rd}$ et $\xi_3 = -0.4 \text{ rd}$ pour l'expérimentation dans le contexte réel. Le pas de calcul utilisé est $dt = 1s$. Ces taux sont des valeurs approximatives des taux de rotation calculés du véhicule dans chaque expérimentation.

Les paramètres des deux expérimentations sont indiqués dans le tableau 3.2. Les paramètres des filtres circulaires associés à chaque modèle du CIMM sont : $\kappa_R = 50$, $\kappa_Q = 1000$. Les matrices de transition sont définies par Π_1 pour la bifurcation en Y et par Π_2 pour le contexte réel.

$$\Pi_1 = \begin{bmatrix} 0.99 & 0.01 \\ 0.01 & 0.99 \end{bmatrix} ; \quad \Pi_2 = \begin{bmatrix} 0.99 & 0.01 & 0.01 \\ 0.01 & 0.99 & 0.001 \\ 0.01 & 0.001 & 0.99 \end{bmatrix} \quad (4.41)$$

4.4.2 Placement des particules par les probabilités de modèle

Le pré-filtrage CIMM fournit deux sorties à l'algorithme de Map matching circulaire particulaire : Une estimation de la direction notée $\hat{\theta}_k$ sur la figure 4.3 et la probabilité de chaque modèle j notée μ_k^j sur la même figure. Ces probabilités sont utilisées par l'algorithme de Map matching pour orienter le placement des particules arrivées à la fin d'une route. Le principe de la méthode utilisée est décrit sur la figure 4.4.

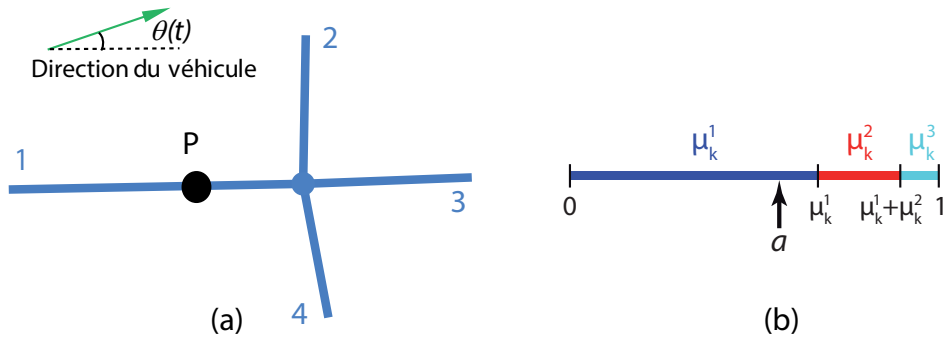


FIGURE 4.4 – Placement des particules en fin de route

Soit la particule P arrivée à la fin de la route 1. La méthode utilisée au chapitre 3 plaçait aléatoirement la particule sur l'une des routes reliées à la route 1. On propose d'améliorer cette méthode en utilisant les probabilités de modèle fournies par le CIMM pour guider le placement des particules sur le bon segment. Sur l'exemple de la figure 4.4.(a), vu la direction du véhicule, il est moins probable que celui-ci emprunte la route 4, la route la plus probable étant la route 3. Cela est reflété par les probabilités des modèles figure 4.4.(b) réparties sous forme de sous-segments de l'intervalle $[0, 1]$, μ_k^1 étant la probabilité de continuer en ligne droite vers la route 3, μ_k^2 la probabilité de tourner à gauche vers la route 2 et μ_k^3 la probabilité de tourner à droite

vers la route 4. On choisit un nombre aléatoire a selon une distribution uniforme entre 0 et 1 : $a \sim \mathcal{U}[0, 1]$. Le modèle m considéré est celui correspondant au sous-segment auquel le nombre a appartient :

$$m = j \text{ tel que } a \in \left[\sum_{s=1}^{j-1} \mu_k^s, \sum_{s=1}^j \mu_k^s \right] \quad (4.42)$$

Cette approche possède l'avantage de préserver la nature statistique de la méthode tout en favorisant le modèle de mouvement le plus probable. Ensuite, l'algorithme recherche sur le réseau routier la route qui correspond au modèle m retenu, soit dans le cas présent la route 3.

4.5 Évaluation de la méthode

4.5.1 Introduction

Afin d'évaluer les performances du filtre IMM circulaire, on reprend les expérimentations réalisées au chapitre 3 sur la bifurcation en Y et la trajectoire dans le contexte réel (voir figure 3.11) en utilisant cette fois un préfiltrage IMM circulaire (CIMM).

On souhaite tout d'abord évaluer l'apport de l'approche IMM sur l'estimation de la direction dans l'expérimentation de la bifurcation en Y illustrée sur la figure 3.11.(a). On évalue, ensuite, la précision de l'estimation de la position sur cette bifurcation en utilisant l'algorithme de Map matching circulaire particulière avec deux types de pré-filtrage : circulaire et IMM circulaire.

L'évaluation de la probabilité de l'identification de la route est faite, par la suite, avec l'expérimentation sur la bifurcation en Y et dans le contexte réel figure 3.11.(b). Dans cette dernière expérimentation, le début et la fin du masquage des points GPS sont choisis aléatoirement.

On évalue aussi la robustesse du Map matching circulaire particulière avec le pré-filtrage CIMM pour différents angles entre les segments de la bifurcation en Y. On garde les mêmes paramètres utilisés au chapitre 3, à savoir une erreur sur la mesure de l'angle distribuée selon un loi de von Mises de paramètre de concentration $\kappa = 30$, une vitesse du véhicule à la bifurcation de 10 km/h, et une erreur sur la mesure de vitesse considérée gaussienne et biaisée de variance 1m/s et un biais distribué selon une loi uniforme entre -0.5 m/s et 0.5 m/s.

Enfin, on évalue les différents algorithmes proposés sur des données réelles recueillies sur

le terrain avec un véhicule équipé d'un récepteur GPS et d'un capteur de vitesse et de direction.

4.5.2 Estimation de la direction

On évalue le pré-filtrage circulaire et CIMM en calculant la dispersion circulaire qui est la moyenne des distances circulaires à chaque instant avec les directions estimées sur 1 000 réalisations du bruit. La figure 4.5.(a) représente la dispersion circulaire en fonction du temps et la figure 4.5.(b) la dispersion circulaire en fonction des réalisations.

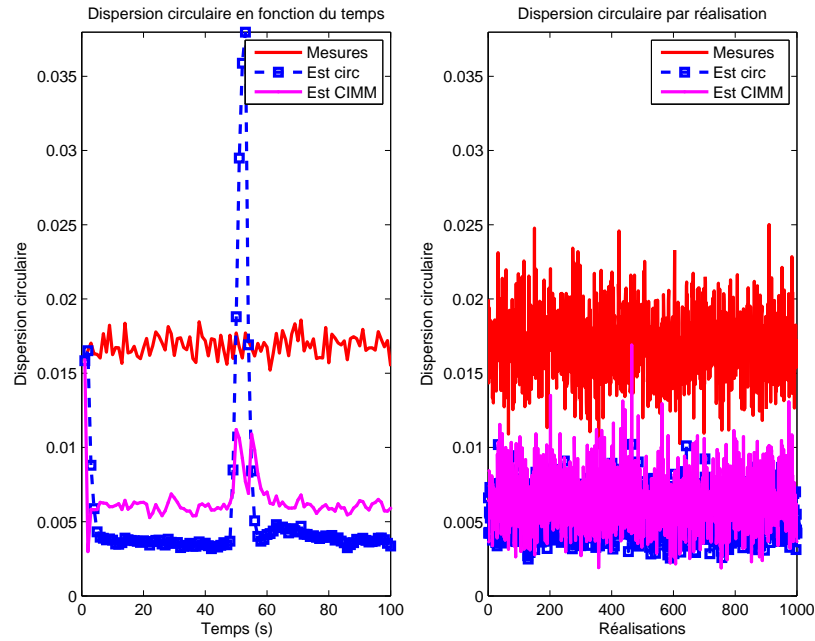


FIGURE 4.5 – Estimation de la direction avec le filtre CIMM

On constate que le pic d'erreur est abaissé au-dessous du bruit de mesure par le CIMM, grâce à l'adoption du modèle avec manœuvres (Modèle 2) sur la bifurcation comme illustré figure 4.6. Le réglage du paramètre κ_R du CIMM impose un compromis entre la réduction du pic d'erreur et la réduction du niveau de bruit, pour un angle de direction constant. Le réglage optimal est obtenu pour $\kappa_R = 50$.

Au niveau de la dispersion circulaire moyenne, elle est quasiment similaire pour les deux filtres, elle est de 0.005 pour le filtre circulaire et de 0.006 pour le filtre CIMM. L'avantage du CIMM est clairement visible à l'instant de la manœuvre où la précision de l'estimation est cruciale, car elle est utilisée dans l'affectation des poids aux particules dans l'algorithme de

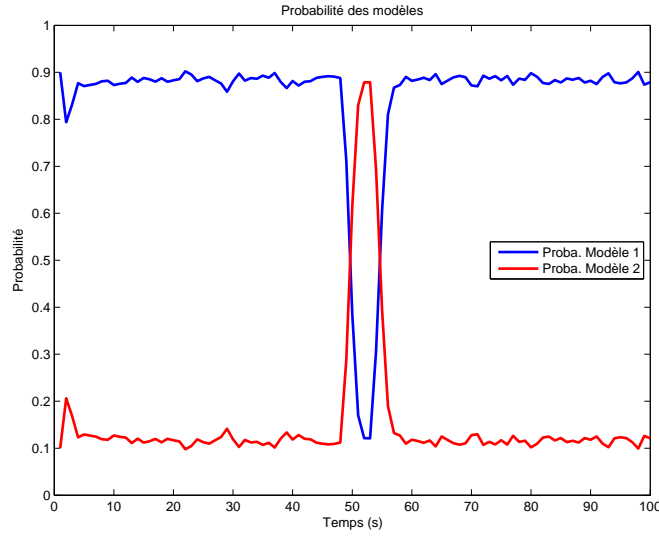


FIGURE 4.6 – Probabilités des modèles du filtre CIMM

Map matching particulière.

4.5.3 Estimation de la position

Dans cette expérimentation, on propose de comparer le Map matching particulière avec pré-filtrage circulaire (noté MAP-SIR-CIR, mis en œuvre au chapitre 3) avec le Map matching particulière avec pré-filtrage CIMM et placement des particules par les probabilités de modèle (noté MAP-SIR-CIMM). On gardera pour la référence l'algorithme topologique noté MAP-TOP.

On utilise la méthode d'évaluation du chapitre 3 qui consiste à calculer la distance moyenne en fonction du temps entre la position réelle et estimée par les différents algorithmes, ainsi que l'erreur moyenne sur les positions estimées en fonction des réalisations et cela pour 1 000 réalisations et un angle de bifurcation $\Delta\alpha = 45^\circ$.

On reporte figure 4.7 les résultats obtenus avec les trois algorithmes. Cette figure montre une amélioration apportée par le filtre MAP-SIR-CIMM à la bifurcation avec, tout d'abord, une réaction en avance de 2s par rapport au MAP-SIR-CIR, ensuite une erreur ramenée à 2.5m contre 3.5m pour le MAP-SIR-CIR au moment de la bifurcation. Ceci est dû à une meilleure re-concentration des particules sur la bifurcation par le MAP-SIR-CIMM grâce à une meilleure estimation de la direction du véhicule et aussi grâce au positionnement des particules par les probabilités de modèle. Ce dernier point sera discuté plus en détails ci-après.

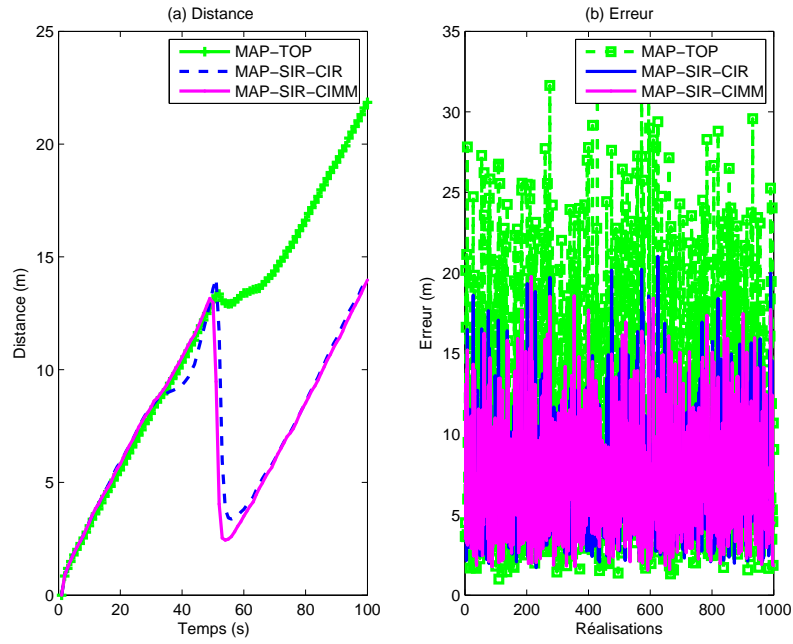


FIGURE 4.7 – Erreur sur la position : Direction filtrée par le CIMM

L'erreur de positionnement globale, qui est la distance moyenne entre les positions réelles et estimées pour 1 000 réalisations de la trajectoire en Y, montre aussi une amélioration avec $7.5m$ pour le MAP-SIR-CIMM, $8.1m$ pour le MAP-SIR-CIR contre $11.2m$ pour le MAP-TOP.

On évalue, ensuite, la robustesse des trois méthodes en calculant l'erreur de positionnement en mètres en fonction de l'angle de la bifurcation $\Delta\alpha$. Le tableau 4.1 montre les résultats obtenus.

$\Delta\alpha$ (°)	45	34	22	11
MAP-TOP	11.2	12.0	14.4	14.8
MAP-SIR-CIR	8.1	8.0	8.7	9.5
MAP-SIR-CIMM	7.5	7.4	7.7	9.2

TABLE 4.1 – Erreur de positionnement globale en fonction de l'angle entre les routes.

Le tableau 4.1 montre bien que l'algorithme MAP-SIR-CIMM est robuste au bruit et reste meilleur en termes de précision jusqu'à un angle $\Delta\alpha = 11^\circ$. Au delà, il n'arrive plus à détecter le changement de modèle avec le niveau de bruit considéré dans l'expérimentation.

Discussion sur le placement des particules

L'algorithme MAP-SIR-CIMM réalise le placement des particules sur les segments en utilisant les probabilités de modèle fournies par le filtre CIMM comme décrit dans la section 4.4.1. L'impact de cette méthode peut être observé sur la figure 4.7.(a) entre les instants 32s et 50s. On note dans cet intervalle une diminution de l'erreur de positionnement pour l'algorithme MAP-SIR-CIR par rapport aux algorithmes MAP-SIR-CIMM et MAP-TOP. Ceci peut être interprété comme suit :

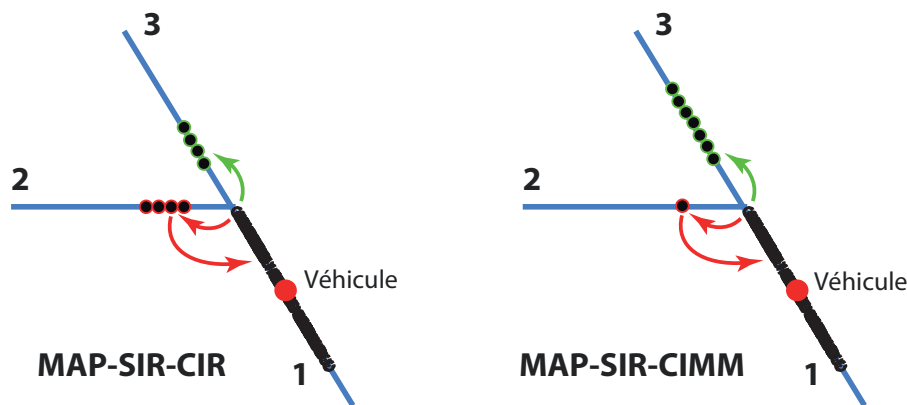


FIGURE 4.8 – Comportement des particules à la bifurcation

Entre 0s et 32s, les trois algorithmes se comportent d'une façon similaire. A partir de 32s, les premières particules, qui sont souvent en avance par rapport à la position réelle, arrivent à la bifurcation (voir figure 4.8). Le MAP-SIR-CIR les place avec une probabilité égale sur les routes 2 (virage) et 3 (pas de virage). Les particules sur la route 2 reçoivent un poids faible car le véhicule n'a pas encore entamé le virage. Par conséquent, au ré-échantillonnage, ces particules sont éliminées puis remplacées par des particules plus probables sur la route 1. Cela augmente la concentration des particules sur la route 1, ce qui se traduit par l'amélioration de la précision à partir de 32s.

L'algorithme MAP-SIR-CIMM se comporte autrement, il place la plupart des particules arrivant à la bifurcation sur la route 3, tant que la manœuvre de rotation n'est pas détectée. Ainsi, l'erreur continue de croître linéairement de la même façon que l'algorithme MAP-TOP.

Afin de mieux cerner ce phénomène, qui peut être perçu de prime abord comme un avantage pour le MAP-SIR-CIR, on a refait l'expérimentation de la bifurcation en Y sans effectuer de virage. On reporte figure 4.9 la distance moyenne en fonction du temps et l'erreur moyenne en

fonction des réalisations pour 1 000 réalisations du bruit.

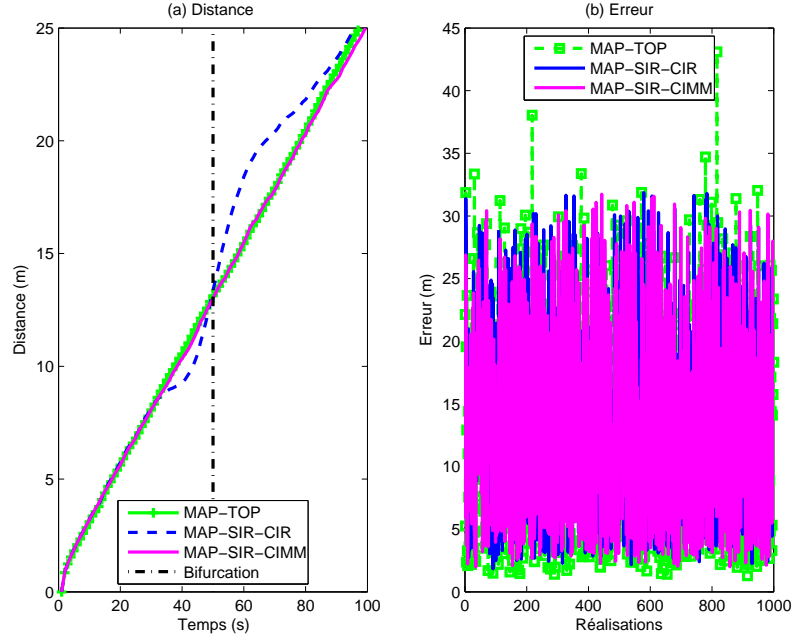


FIGURE 4.9 – Erreur sur la position : expérimentation sans virage

On observe, juste après la bifurcation, une augmentation de l'erreur jusqu'à $3m$ pour l'algorithme MAP-SIR-CIR et pour une durée de $40s$ environ. L'origine de ce pic d'erreur provient de la re-concentration des particules sur la route 1 sous l'effet du placement aléatoire des particules comme expliqué précédemment. Si ce phénomène améliore l'estimation juste avant que le véhicule effectue le virage, il détériore cette précision si le véhicule ne tourne pas. En effet dans ce cas, il provoque le recul de la position estimée par le MAP-SIR-CIR par rapport à la position réelle et entraîne donc l'apparition de cette augmentation de l'erreur de positionnement qui s'estompe avec la redistribution progressive des particules.

Le MAP-SIR-CIMM évite ce phénomène en considérant les probabilités de modèle. Dans ce cas, et puisque le modèle indiqué par le CIMM est toujours le modèle 1 sans manœuvre, la quasi-totalité des particules sont propagées de la route 1 vers la route 3 garantissant une évolution uniforme de l'erreur de positionnement au même titre que le MAP-TOP.

Globalement, dans cette expérimentation sans virage, le MAP-SIR-CIMM reste meilleur avec une erreur de positionnement moyenne de $13.3m$ contre $14.2m$ pour le MAP-SIR-CIR et $13.5m$ pour le MAP-TOP.

4.5.4 Identification de la route

Evaluation avec la trajectoire en Y

On désire évaluer la probabilité d'identifier correctement la route sur laquelle se trouve le véhicule. La figure 4.10 montre cette probabilité évaluée en fonction du temps à partir de 1 000 réalisations de la trajectoire.

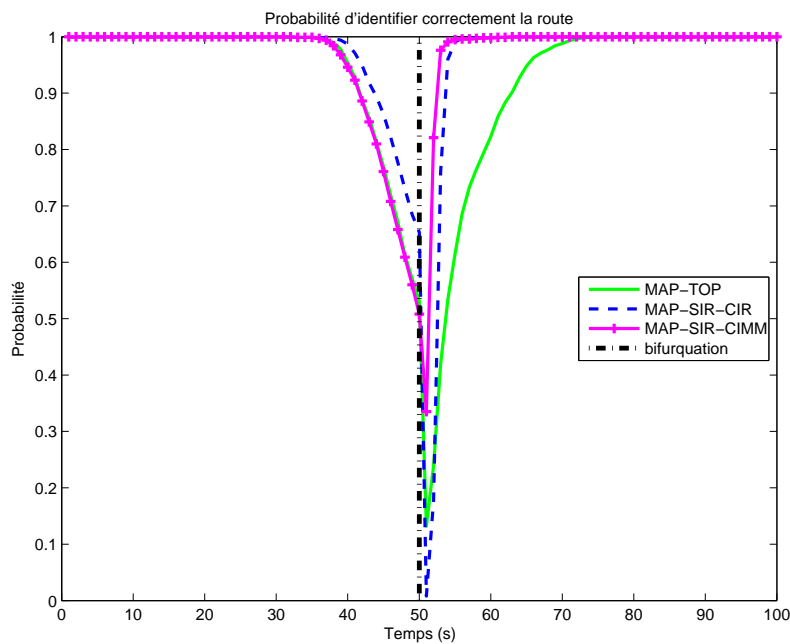


FIGURE 4.10 – Probabilité d'identifier correctement le segment de route

On remarque que l'algorithme MAP-SIR-CIMM est moins performant que le MAP-SIR-CIR avant la jonction et que son comportement est similaire au MAP-TOP. Ce résultat se comprend par le placement de la position estimée par MAP-SIR-CIMM tantôt sur la route initiale 1 et tantôt sur la route colinéaire 3 avant la manœuvre.

En revanche, et grâce à sa réaction à la détection de la manœuvre, le MAP-SIR-CIMM identifie la nouvelle route plus rapidement. On observe un minimum de probabilité de détection de 0.33 qui passe à 0.98 après 3s de la manœuvre. Le MAP-SIR-CIR est moins précis et moins rapide avec un minimum de probabilité de détection nul et un passage à 0.98 de probabilité après 4.5s de la manœuvre. Globalement le MAP-SIR-CIMM est légèrement meilleur avec une probabilité moyenne d'identification de 0.956 contre 0.943 enregistré pour la MAP-SIR-CIR.

Afin d'évaluer les performances globales d'identification sur la jonction en Y, on calcule la

probabilité d'identifier la bonne route sur toute la trajectoire. On reporte sur la tableau 4.2 cette probabilité en fonction de la différence d'angles $\Delta\alpha$ entre les routes de la jonction.

$\Delta\alpha$ (°)	45	34	22	11
MAP-TOP	0.915	0.895	0.801	0.772
MAP-SIR-CIR	0.943	0.946	0.937	0.926
MAP-SIR-CIMM	0.956	0.954	0.949	0.908

TABLE 4.2 – Probabilité d'identification en fonction de l'angle entre les routes.

On constate que l'algorithme MAP-SIR-CIMM améliore la probabilité d'identification sauf pour l'angle 11° où le CIMM n'arrive plus à différencier les deux modèles à cause de l'affaiblissement du rapport signal à bruit.

Dans l'expérimentation faite sans effectuer le virage, on reporte figure 4.11 la probabilité d'identifier correctement la route pour les trois algorithmes avec 1 000 réalisations de la trajectoire en Y.

On observe les mêmes évolutions de la probabilité d'identification que dans l'expérimentation

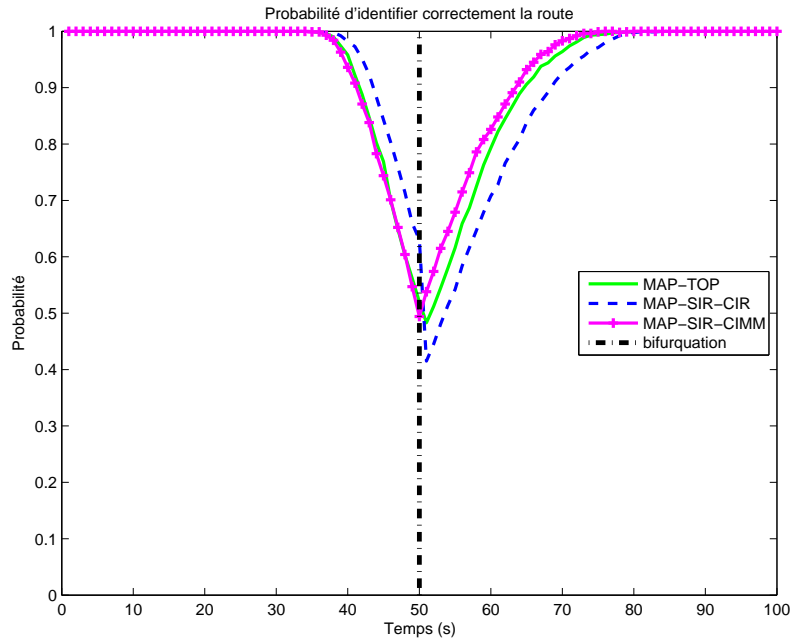


FIGURE 4.11 – Probabilité d'identifier correctement le segment de route - Sans virage

avec virage. Le MAP-SIR-CIMM est moins bon avant la bifurcation, mais il est meilleur après. Dans cette expérimentation aussi l'algorithme MAP-SIR-CIMM est meilleur avec une probabilité globale sur toute la trajectoire de 0.931, lorsque celle-ci est de 0.918 pour le MAP-SIR-CIR

et de 0.925 pour le MAP-TOP.

Evaluation dans le contexte réel

On considère la trajectoire décrite au chapitre 3 sur la figure 3.16, reconstruite à l'aide de données réelles. On considère un véhicule équipé d'un récepteur GPS, d'un odomètre pour la mesure de vitesse et d'un magnétomètre pour la mesure de direction. La fréquence de mesure des capteurs proprioceptifs est de 16 mesures/s, et 1 mesure/s pour le GPS. Les erreurs capteurs sont calculées à partir des données réelles, et l'on considère une erreur gaussienne d'un écart type de 5% sur la valeur mesurée de la vitesse et un biais de 5% sur la mesure de vitesse distribué selon une loi uniforme. L'erreur sur la direction est modélisée par une distribution de von Mises de paramètre de concentration de 30. L'erreur GPS est définie comme la variance de la distance entre la position mesurée et la position réelle.

La première expérimentation dans le contexte réel consiste à comparer les performances des algorithmes MAP-SIR-CIR et MAP-SIR-CIMM lorsque les signaux GPS sont masqués sur toute la trajectoire sauf le premier point, tandis qu'ils sont disponibles pour l'algorithme MAP-TOP avec des précisions de positionnement différentes. La figure 4.12 montre, pour les trois méthodes, l'estimation du pourcentage de segments identifiés correctement obtenue avec 100 réalisations de la trajectoire. Ce pourcentage est estimé pour différentes précisions GPS.

Les résultats montrent que l'algorithme MAP-SIR-CIMM identifie en moyenne 96% des segments correctement, alors que ce taux d'identification est de 93.6% pour l'algorithme MAP-SIR-CIR. En comparaison avec l'algorithme topologique, on constate que le MAP-SIR-CIMM devient plus performant que le MAP-TOP dès que l'erreur GPS dépasse 4m. Le MAP-SIR-CIR ne dépasse en performance le MAP-TOP qu'à partir d'une erreur GPS de 7.5m.

La seconde expérimentation conduite dans le contexte réel consiste à masquer n points GPS à un point de départ choisi aléatoirement sur la trajectoire. On génère 100 réalisations pour chaque valeur de n . On compare les trois algorithmes qui utilisent les données de vitesse et de la direction pendant le masquage. Sur le tableau 4.3, on reporte le pourcentage des segments correctement identifiés en fonction du pourcentage de points masqués pour une erreur GPS de 10m. La figure 4.13 montre l'évolution du pourcentage des segments identifiés correctement en fonction du pourcentage des points masqués pour des erreurs GPS de 0m et de 10m.

On constate que l'algorithme MAP-SIR-CIMM identifie plus de segments corrects pendant

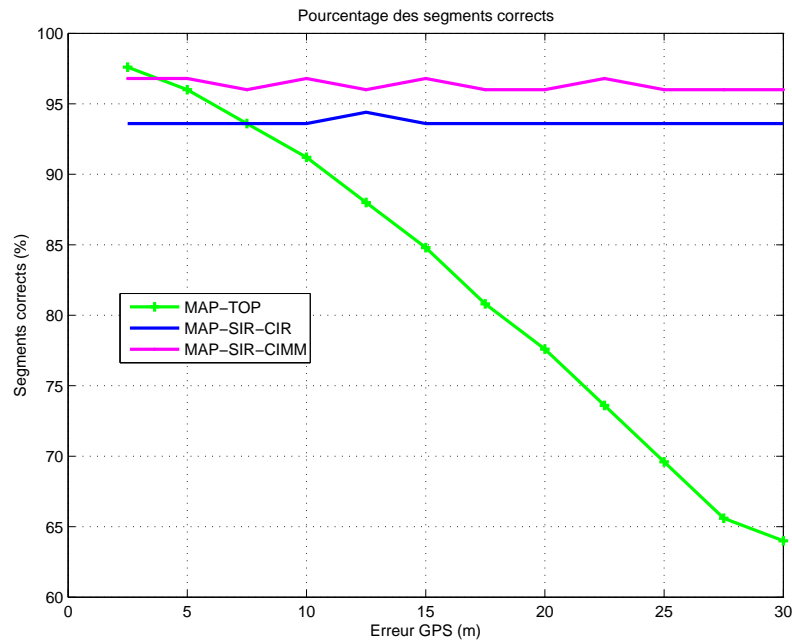


FIGURE 4.12 – Pourcentage des segments correctement identifiés pour différentes précisions GPS

Points masqués	8%	24%	40%	56%	72%
MAP-TOP	91.2%	91.2%	90.4%	89.6%	88.8%
MAP-SIR-CIR	91.2%	91.5%	91.2%	91.5%	92.6%
MAP-SIR-CIMM	91.1%	91.8%	92.6%	93.7%	94.8%

TABLE 4.3 – Pourcentage de segments corrects en fonction du pourcentage de points masqués (Erreur GPS = 10m)

le masquage par rapport aux algorithmes MAP-SIR-CIR et MAP-TOP. Cette performance est d'autant plus marquée que la durée de masquage est plus importante et que l'erreur GPS est élevée. On conclut que le MAP-SIR-CIMM est plus performant en cas de masquage des signaux GPS, même pour un récepteur GPS sans erreur comme indiqué sur la figure 4.13.

4.5.5 Application aux données réelles

Contexte expérimental

L'expérimentation avec les données réelles a été réalisée à l'aide d'un véhicule instrumenté de marque Renault Kangoo équipée de plusieurs capteurs reliés à un système d'acquisition et de synchronisation de marque HBM (voir figure 4.14). On a utilisé dans notre expérimentation

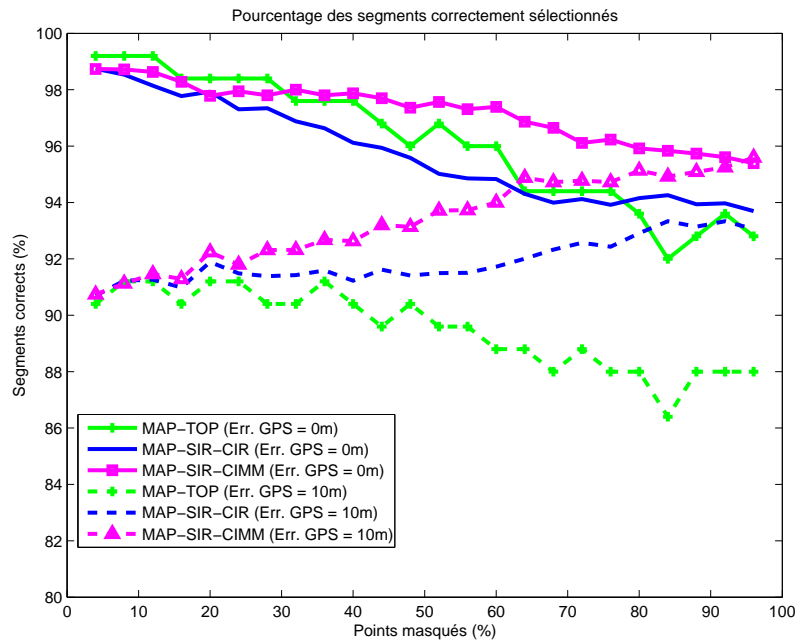


FIGURE 4.13 – Pourcentage des segments correctement identifiés en fonction du pourcentage de points masqués

un magnétomètre de marque Crossbow CXM543 pour la mesure de la direction du véhicule, un capteur de vitesse optique de marque CORREVIT L-400 pour la mesure de la vitesse, et un récepteur GPS de marque Garmin pour la mesure de position.

La trajecte, illustrée sur la figure 4.15, a été réalisée dans la ville de Calais en France le 11 juillet 2014. Le départ était à 11 : 36 : 25 rue des Frontinettes et l'arrivée était à 11 : 50 : 45 sur le parking de l'université du Littoral Côte d'Opale. La durée du trajet est de 14min 20s et la distance parcourue de 5.6km. Le véhicule traverse 3 rond-points numérotés de (1) à (3) sur la figure 4.15. Il réalise une rotation complète sur le rond-point (1). Il marque deux arrêts sur deux feux rouges : le premier noté (A) d'une durée de 68s et le deuxième noté (B) d'une durée de 34s.

La fréquence des mesures de position par le récepteur GPS est de 0.5Hz, les mesures de vitesse et de direction sont obtenues avec une fréquence de 2Hz. Le trajet comporte 1720 mesures de direction et de vitesse et 430 points GPS.

La figure 4.16 montre la trajectoire de référence obtenue par les mesures GPS ainsi que la trajectoire obtenue par la localisation à l'estime où l'on calcule chaque position à partir de la position précédente et des données de vitesse et de direction (cf. équations (3.1) et (3.2)). On remarque clairement que l'écart entre les deux trajectoires devient de plus en plus impor-

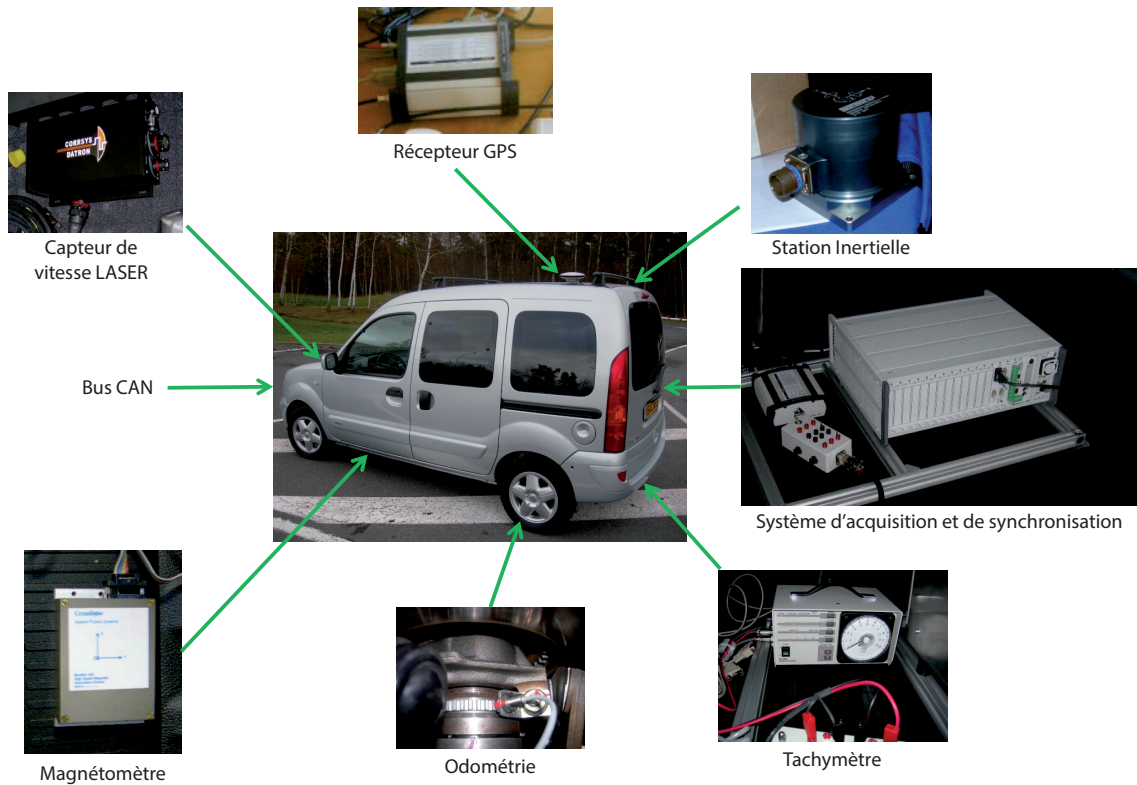


FIGURE 4.14 – Voiture instrumentée utilisée dans l'expérimentation

tant avec le temps à cause des incertitudes sur la vitesse et sur la direction. La localisation par l'estime est utilisée par l'algorithme topologique MAP-TOP en cas de masquage, le point de départ étant le dernier point GPS avant le masquage des signaux des satellites GPS. Les deux algorithmes MAP-SIR-CIR et MAP-SIR-CIMM utilisent les données de vitesse et de direction pour propager les particules sur les différentes routes comme décrit dans la section 3.3.3. L'algorithme MAP-SIR-CIR effectue un préfiltrage de la direction par un filtre circulaire, alors que l'algorithme MAP-SIR-CIMM utilise un filtre IMM circulaire pour le préfiltrage en plus du placement des particules par les probabilités des modèles dont le principe est décrit dans la section 4.4.1.

Evaluation des méthodes

On évalue les performances des algorithmes de Map matching particulière circulaire MAP-SIR-CIR et MAP-SIR-CIMM par rapport à l'algorithme de Map matching topologique MAP-

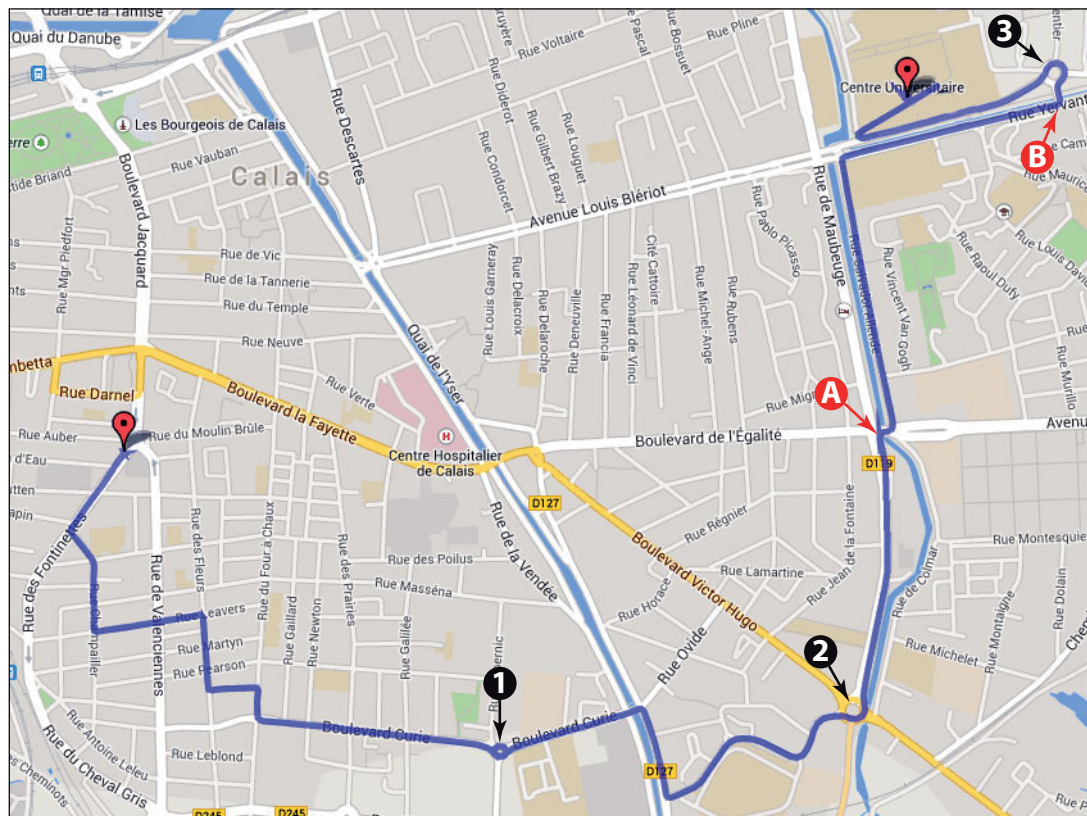


FIGURE 4.15 – Trajet réalisé avec la voiture instrumentée

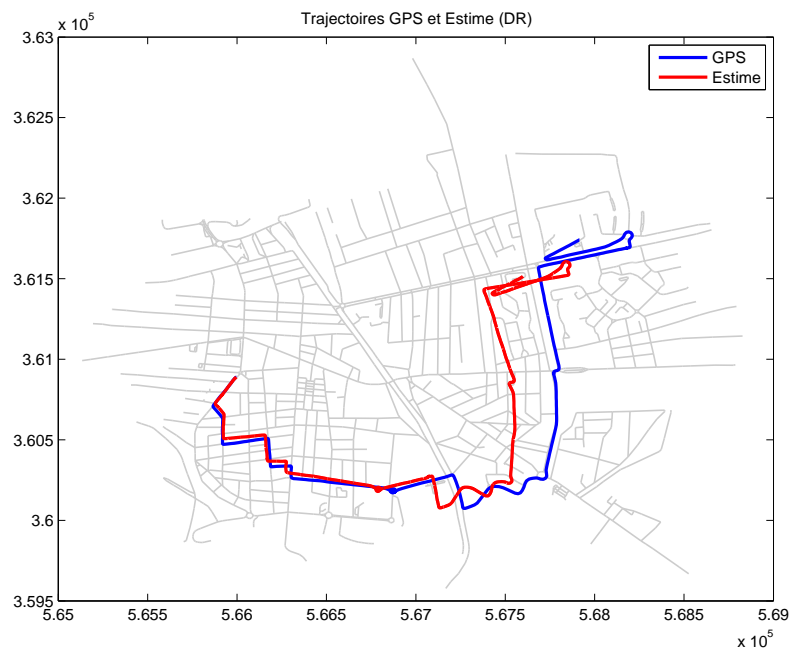


FIGURE 4.16 – Trajectoire avec données réelles - GPS et estime

TOP. Les paramètres des filtres sont les suivants :

- **Algorithme MAP-TOP** : On garde la valeur des paramètres indiqués dans le tableau 3.2.
- **Filtre circulaire de l'algorithme MAP-SIR-CIR** : $\kappa_Q = 1e3$, $\kappa_{\tilde{Q}} = 1e4$, $\kappa_R = 800$. Ce dernier paramètre a été augmenté par rapport à l'expérimentation dans le contexte réel car le bruit sur l'angle mesuré est plus faible.
- **Filtre CIMM de l'algorithme MAP-SIR-CIMM** : On considère les mêmes modèles et les mêmes paramètres des filtres que dans le contexte réel décrit dans la section 4.4.1. Seuls les taux de rotation considérés pour les modèles 2 et 3 sont modifiés à $\xi_2 = 0.2 \text{ rd}$ et $\xi_3 = -0.2 \text{ rd}$ car la fréquence des mesures du magnétomètre est de $2Hz$.

Pour les deux algorithmes MAP-SIR-CIR et MAP-SIR-CIMM, on a considéré un paramètre de concentration $\kappa_{R_p} = 1$ pour la distribution de von Mises utilisée dans l'affectation des poids aux particules. Ce paramètre est plus petit que celui utilisé dans les expérimentations précédentes (voir tableau 3.2). L'avantage est de permettre aux particules les moins probables de continuer à survivre plus longtemps et leur permettre ainsi d'explorer plus de routes.

Les mesures de vitesse et de direction ainsi que les probabilités de modèles calculées par le filtre CIMM sont illustrées sur la figure 4.17. On constate que les probabilités de modèle reflètent bien les différentes manœuvres du véhicule et particulièrement la rotation sur les ronds-points (1), (2) et (3) aux intervalles de temps : $274s - 300s$, $444s - 452s$ et $766s - 780s$ respectivement. Ces manœuvres sont indiquées par une probabilité qui avoisine 1 pour le modèle 2. On constate aussi une transition entre $-\pi$ et π pendant la rotation sur les ronds-points (1) et (3) qui ne perturbe pas le filtre circulaire CIMM.

On désire évaluer les performances de chaque algorithme en fonction de la durée de masquage des données GPS. La méthode d'évaluation proposée consiste à simuler le masquage des signaux GPS sur n points de la trajectoire de référence. Étant donné la fréquence de mesure GPS, un point masqué correspond à une durée de 2 secondes. On génère 100 points de départ choisis aléatoirement pour chaque nombre n de points masqués consécutivement. On évalue ensuite le pourcentage des segments correctement identifiés par chaque algorithme ainsi que la moyenne de la distance entre la position GPS et la position estimée par l'algorithme. En dehors des zones de masquage, le Map matching topologique MAP-TOP est utilisé par tous les algorithmes.

On reporte sur le tableau 4.4 et la figure 4.18 le pourcentage des segments correctement identifiés pour les trois algorithmes en fonction du pourcentage de points masqués sur la trajectoire.

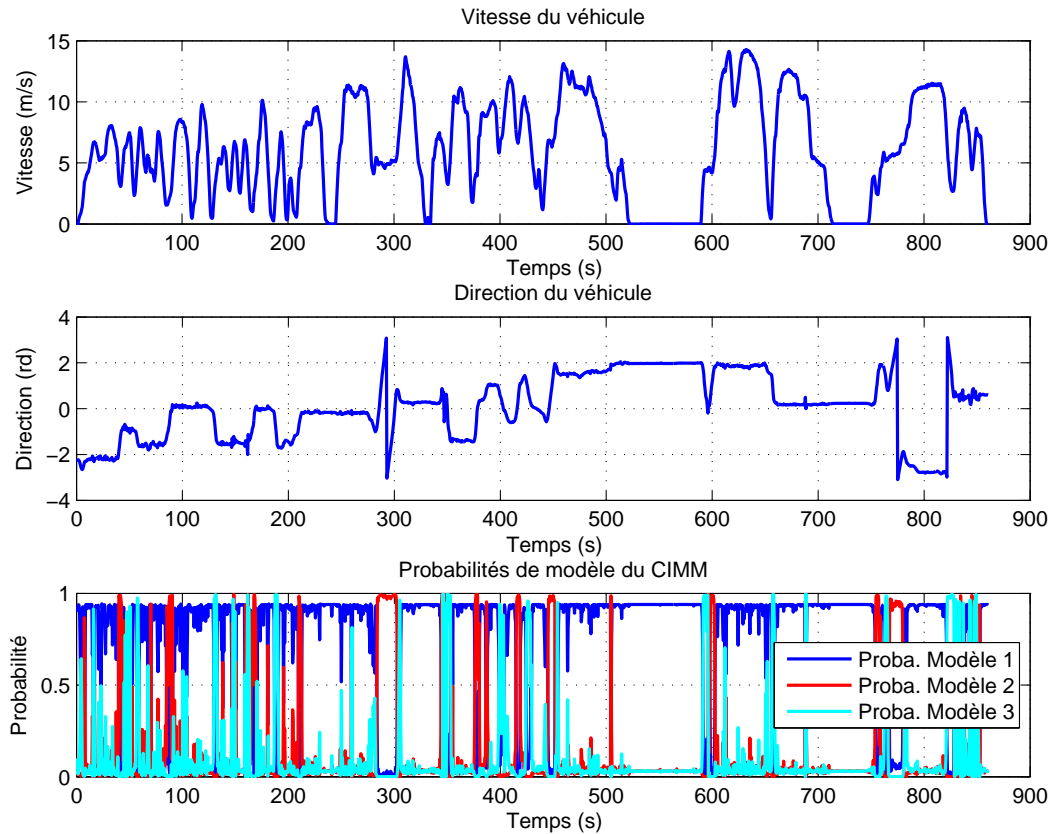


FIGURE 4.17 – Mesures de vitesse, de la direction et probabilités de modèle du CIMM

Points masqués (%)	6	23	41	58	76	99
MAP-TOP (%)	98.0	83.8	68.2	50.7	38.5	17.8
MAP-SIR-CIR (%)	98.4	92.4	85.0	73.6	67.4	55.5
MAP-SIR-CIMM (%)	98.5	92.4	87.6	79.7	75.4	69.7

TABLE 4.4 – Pourcentage de segments corrects en fonction du pourcentage des points masqués

On constate une rapide détérioration des résultats de l'algorithme MAP-TOP, tandis que l'algorithme MAP-SIR-CIMM garde les meilleures performances avec un pourcentage de segments correctement identifiés au voisinage de 70%, lorsque les signaux GPS sont masqués durant 99% de la trajectoire, soit presque 14 minutes. L'algorithme MAP-SIR-CIR est moins performant car son taux d'identification des segments corrects ne dépasse pas 56% pour la même durée de masquage.

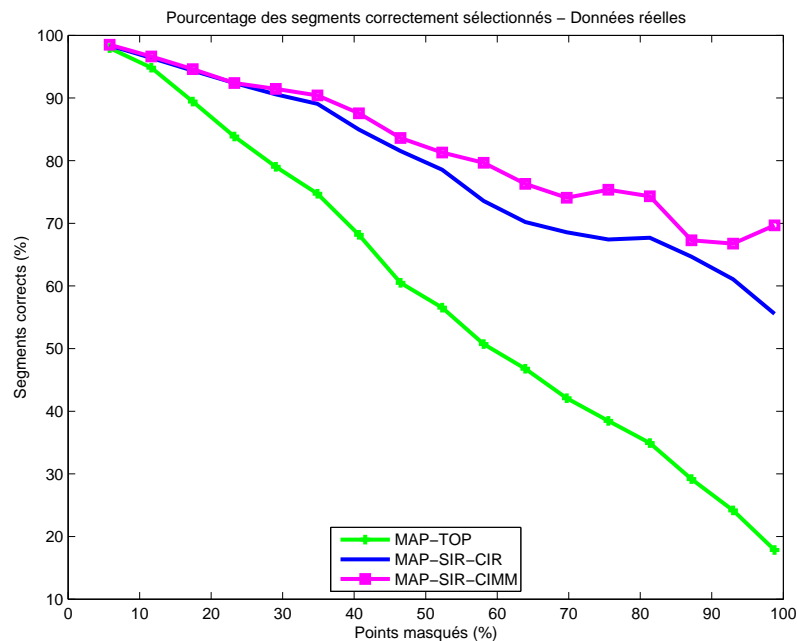


FIGURE 4.18 – Pourcentage des segments correctement identifiés en fonction du pourcentage des points masqués - Données réelles

L'évaluation qui porte sur la distance moyenne entre la position GPS et la position estimée, dont les résultats sont reportés sur le tableau 4.5 et la figure 4.19, montre aussi une meilleure précision de l'algorithme MAP-SIR-CIMM. En effet cette distance moyenne est proche de $18m$ lorsque 99% de la trajectoire est masquée, tandis qu'elle est de $23.7m$ pour l'algorithme MAP-SIR-CIR et de $192m$ pour l'algorithme MAP-TOP.

Points masqués (%)	6	23	41	58	76	99
MAP-TOP (m)	1.0	16.2	43.0	86.0	136.6	191.9
MAP-SIR-CIR (m)	2.2	6.7	12.6	17.6	19.7	23.7
MAP-SIR-CIMM (m)	1.8	6.6	10.2	13.4	16.1	18.1

TABLE 4.5 – Distance moyenne entre position GPS et estimée en fonction du pourcentage de points masqués

Discussion

Un avantage important de l'algorithme MAP-SIR-CIMM par rapport à l'algorithme topologique classique est observé dans le cas ambigu de deux routes parallèles comme illustré figure 4.20. Le véhicule qui se déplace sur la route 2 fait un virage vers la route 4. La trajectoire cal-

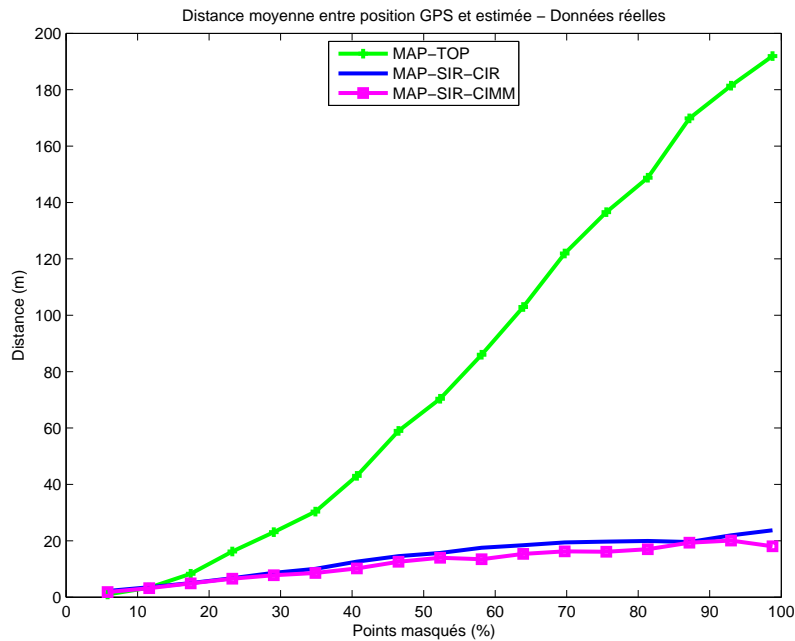


FIGURE 4.19 – Distance moyenne entre position GPS et estimée en fonction du pourcentage des points masqués - Données réelles

culée par la localisation par l'estime se trouve entre les deux routes parallèles 3 et 4. A l'étape (b), l'algorithme topologique MAP-TOP choisit par erreur la route 3 qui est la plus proche de la position localisée par l'estime. L'algorithme MAP-SIR-CIMM, en revanche, place des particules sur les deux routes 3 et 4 car elles ont des probabilités quasiment égales, vu leur direction comparée à la direction de mouvement. La position estimée dépend de la route sur laquelle le maximum de particules existe (dans notre cas c'est la route 4). Les deux groupes de particules continuent d'évoluer indépendamment sur les deux routes jusqu'à ce que le véhicule effectue un virage vers la route 5 (étape (c)). Puisqu'il n'est pas possible d'effectuer un virage sur la route 3, les particules sur cette route se trouvent affectées d'un poids faible et sont aussitôt éliminées et remplacées par de nouvelles particules sur la route correcte 5.

Dans cette expérimentation on observe cependant dans certains cas une "pénurie" des particules qui est d'autant plus fréquente que le temps de masquage est long. Cette pénurie met en défaut les algorithmes MAP-SIR-CIR et MAP-SIR-CIMM.

Le problème de la pénurie des particules survient lorsque toutes les particules se retrouvent sur des routes incorrectes. Après plusieurs itérations, ces particules reçoivent des poids faibles car les mouvements du véhicule ne correspondent pas aux routes considérées, ce qui provoque l'élimination progressive de toutes les particules. Sur les 100 réalisations faites pour chaque

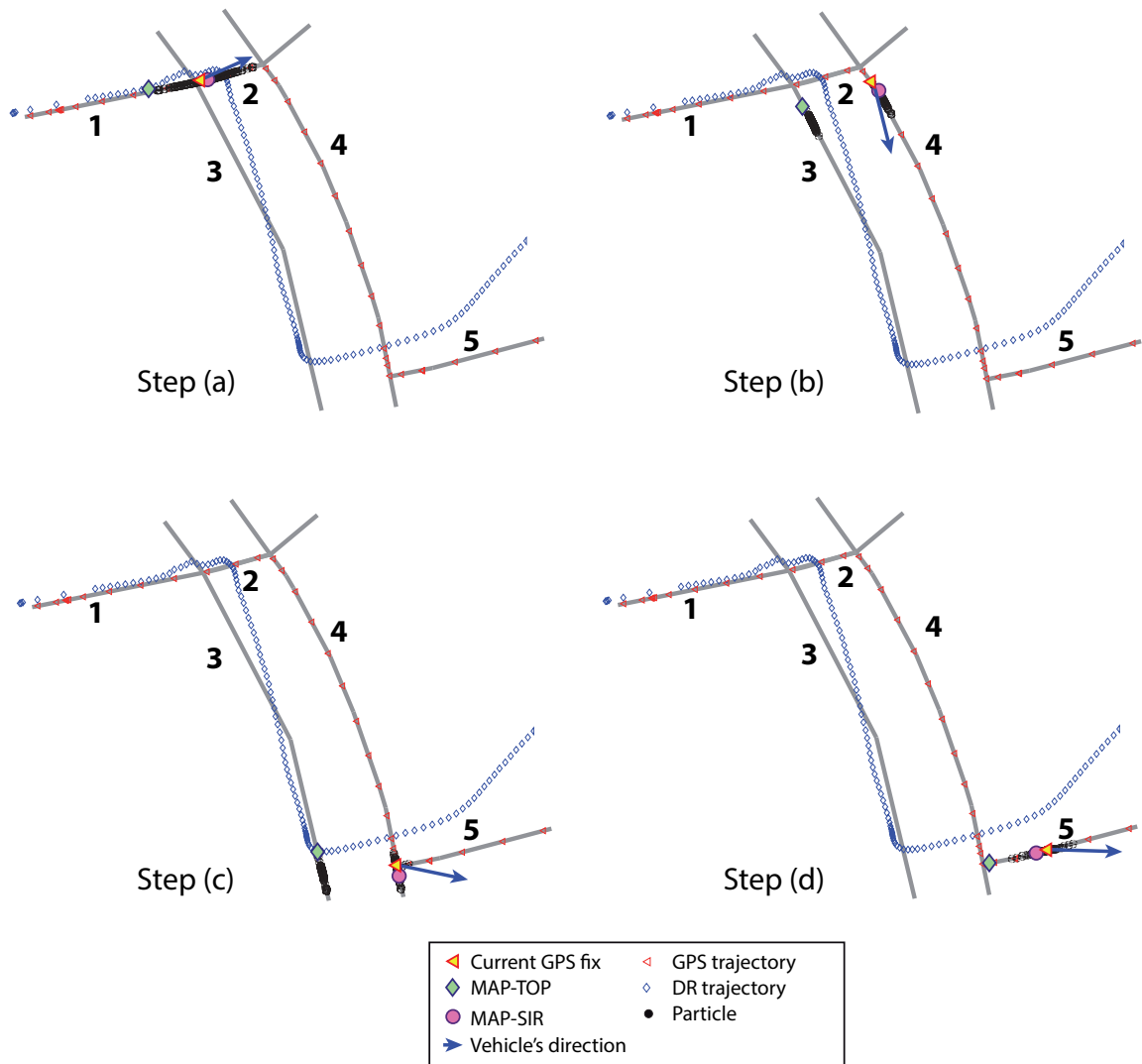


FIGURE 4.20 – Comportement des algorithmes en cas de routes parallèles

nombre n de points masqués, nous avons relevé le pourcentage des réalisations où l'un ou les deux algorithmes se bloquent à cause de la pénurie de particules. Les résultats sont présentés sur le tableau 4.6 et la figure 4.21.

Points masqués (%)	6	17	29	41	52	64	76	87	99
MAP-SIR-CIR (%)	0	1	5	14	23	35	31	28	40
MAP-SIR-CIMM (%)	0	1	4	7	17	22	18	26	21

TABLE 4.6 – Pourcentage des réalisations avec pénurie des particules en fonction du pourcentage de points masqués

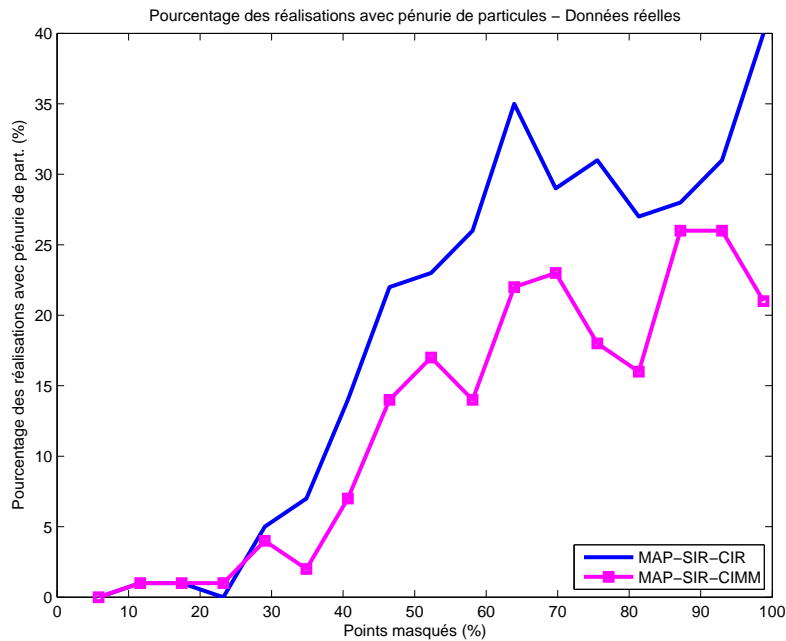


FIGURE 4.21 – Pourcentage des réalisations avec pénurie de particules en fonction du pourcentage de points masqués - Données réelles

D'une manière générale, l'algorithme MAP-SIR-CIMM se bloque moins fréquemment que le MAP-SIR-CIR, et ceci grâce au placement des particules avec les probabilités de modèles qui permet d'orienter plus de particules vers les segments les plus probables.

La pénurie de particule survient lorsqu'on "perd la trace" du véhicule. Dans ce cas la position réelle ne se trouve plus dans l'espace couvert par les particules. Ainsi lorsque le véhicule effectue un virage (en fin de segment), aucune particule ne peut suivre le véhicule dans sa manœuvre et on "perd la trace" du véhicule. En effet dans ce cas les particules peuvent ne pas être arrivées en fin de segment (retard du nuage de particule sur la position réelle) ou avoir toutes dépassé la fin du segment (avance des particules sur la position réelle).

La première raison pour laquelle on observe une pénurie des particules est liée au modèle d'erreur capteur. On remarque notamment cette pénurie lorsque le véhicule se déplace sur de longues distances sans virage. Deux problèmes peuvent alors survenir. Le premier problème est un modèle d'erreur capteur trop faible et la position réelle n'est plus couverte par le nuage de particules. Le second problème est une dispersion trop grande des particules (nombre de particules trop faible) qui introduit un retard à la prise en compte d'une manœuvre. Suite à ce retard l'erreur de positionnement peut être importante et la position réelle peut ne plus être couverte par le nuage de particules.



FIGURE 4.22 – Vectorisation d'un rond-point

La deuxième raison pour laquelle on observe une pénurie des particules est liée aux erreurs engendrées par la vectorisation des routes. La figure 4.22 montre l'image satellitaire du premier rond point de la trajectoire et les segments qui lui correspondent après vectorisation. Le rond point n'est pas un cercle mais plutôt une association de segments linéaires et les segments ne sont pas toujours alignés avec les routes réelles. L'accès du véhicule au rond point se fait dans la réalité d'un façon "douce", alors que sur la carte, les routes donnant accès au rond point forment un angle de 90° avec celui-ci. Une autre source d'erreur est la non prise en considération de la largeur des routes et la possibilité du déplacement du véhicule d'une voie à l'autre dans les routes à double voie. Toutes ces erreurs provoquent inévitablement un écart grandissant avec le temps entre la position estimée et la position réelle.

Ce phénomène de pénurie des particules peut être atténué par l'augmentation du bruit de commande sur les particules et le nombre de particules. Cependant dans ce cas, le coût calcul de l'algorithme est augmenté.

4.6 Conclusion

Dans ce chapitre, nous avons présenté une amélioration de l'algorithme de Map matching particulière circulaire proposé au chapitre 3 par l'application de l'approche multi-modèles. En effet, nous avons observé que le filtre circulaire ne réagissait pas assez rapidement aux manœuvres du véhicules dans les virages, ce qui a entraîné l'apparition de pics d'erreurs sur la

direction estimée. L'approche multi-modèles apporte une solution à ce problème par le choix du modèle approprié au comportement du véhicule à chaque instant.

L'algorithme multi-modèles appliqué au préfiltrage de la direction du véhicule est l'IMM associé à des filtres circulaires (noté CIMM), cela est justifié par la nature circulaire des observations. Deux types de modèles ont été adoptés : un modèle sans manœuvre et un ou deux modèles avec manœuvres (rotation à gauche et/ou à droite). L'algorithme CIMM a permis d'améliorer l'estimation de la direction pendant les manœuvres tout en gardant un niveau d'erreur proche du filtre circulaire en absence de manœuvre.

L'algorithme de Map matching étudié au chapitre 3 est repris dans ce chapitre avec deux améliorations : d'une part l'utilisation d'une estimation de direction plus précise fournie par le filtre CIMM et d'autre part un placement amélioré des particules sur les fins de routes grâce aux probabilités de modèles du filtre. Le principe de ce placement consiste à propager les particules aux bifurcations en prenant en compte le comportement du véhicule qui est reflété par les probabilités des modèles du filtre CIMM. Cet algorithme noté MAP-SIR-CIMM a montré de meilleures performances tant au niveau de la distance moyenne entre la position réelle et estimée, qu'au niveau du pourcentage des segments correctement identifiés sur les trajectoires étudiées.

L'expérimentation menée sur un trajet défini dans un contexte réel a montré que l'algorithme MAP-SIR-CIMM proposé est plus performant que l'algorithme topologique MAP-TOP dès que l'erreur de positionnement GPS dépasse $4m$. Dans cette expérimentation les points GPS sont masqués pour l'algorithme MAP-SIR-CIMM et ne sont pas masqués pour l'algorithme topologique MAP-TOP.

L'algorithme proposé a été évalué sur une trajectoire réelle parcourue par une voiture équipée d'un capteur de vitesse (capteur optique), d'un capteur de direction (magnétomètre) et d'un capteur GPS. L'évaluation des performances de l'algorithme sur ce trajet de $5.6km$ et de durée 14 minutes a permis de montrer la supériorité de l'algorithme MAP-SIR-CIMM comparé à l'algorithme topologique MAP-TOP. En effet pour l'algorithme proposé et en cas de masquage, le pourcentage de segments correctement identifiés est supérieur, et la distance moyenne entre la position estimée et la position GPS est plus faible. Le pourcentage d'identification correcte des segments est proche de 70% pour un masquage du GPS sur la totalité de la trajectoire.

Toutefois, l'algorithme MAP-SIR-CIMM a montré des limitations pour des durées de mas-

quage prolongées en absence de virage. En effet, l'écart entre la position estimée et réelle augmente au cours du temps à cause de l'accumulation des erreurs. Les origines de ces erreurs sont d'une part les imprécisions des capteurs et d'autre part la vectorisation des routes sur la carte qui ne reflète pas la réalité du réseau routier. Ces erreurs introduisent un écart entre la position réelle et la position estimée. L'accumulation de ces écarts a pour conséquence que la position réelle n'appartienne plus au nuage de particules. Dans ce contexte les particules finissent par disparaître totalement et l'algorithme se bloque. Ce phénomène peut être atténué par l'augmentation du bruit de commande sur les particules et le nombre de particules. Cependant dans ce cas le coût calcul de l'algorithme est augmenté.

Conclusion générale

Le travail de recherche présenté dans cette thèse porte sur la localisation d'un objet mobile dans un environnement contraint où les satellites GPS sont fréquemment masqués. Une nouvelle approche de Map matching est proposée pour améliorer la localisation durant les périodes de masquage en utilisant des mesures proprioceptives et des mesures de cap. Cette approche est définie dans un cadre bayésien par une modélisation statistique circulaire.

Ce mémoire est composé de quatre chapitres. Le premier chapitre expose la problématique du map matching puis traite les différentes techniques qui y sont employées. Les principes exposés vont des simples méthodes géométriques jusqu'aux méthodes avancées utilisant la logique floue, la théorie de croyance ou l'analyse probabiliste. Une attention particulière est donnée à la description de l'algorithme topologique de Quddus qui est l'approche de référence pour le map matching.

Le second chapitre est consacré à la description de l'estimation multi-modèle où il s'agit d'estimer l'état d'un système en faisant appel à un ensemble de filtres associés à des modèles simples. Chacun d'entre eux est sensé représenter un mode de fonctionnement ou une dynamique différente du système. L'avantage d'une telle approche est de remplacer un modèle complexe ou non trivial par un ensemble de modèles élémentaires plus faciles à implémenter. L'approche multi-modèle est particulièrement intéressante dans la poursuite des cibles manœuvrantes où elle offre plus de robustesse durant les manœuvres. Une attention particulière est donnée dans ce chapitre à la description de l'approche IMM qui sera redéfinie au chapitre 4 dans le domaine circulaire pour l'estimation du cap du véhicule.

Le troisième chapitre présente l'architecture du système de Map matching que nous proposons. Ce système se distingue des algorithmes classiques par l'utilisation d'un filtre particulière SIR. Ce filtre met en œuvre les mesures de vitesse et de direction non pas pour calculer une position unique comme dans la navigation par l'estime, mais plutôt pour propager des particules sur toutes les routes possibles. On arrive ainsi à mieux compenser les erreurs des capteurs pro-

prioceptifs et à assurer un meilleur positionnement pendant les masquages des satellites GPS. On montre aussi dans ce chapitre qu'en présence de bruit, le préfiltrage des observations de direction par un filtre circulaire améliore la précision de l'algorithme de Map matching. Les expérimentations menées sur des données synthétiques montrent la performance de notre méthode de Map matching ainsi que sa robustesse dans le cas problématique connu de la bifurcation en Y. Dans un contexte réel, l'algorithme de Map matching proposé assure une meilleure identification des segments de route corrects pour des durées de masquage variables. Un constat important déduit à partir de ces expérimentations montre que la méthode de Map matching particulière utilise les virages comme des repères pour re-concentrer les particules et obtenir un gain substantiel en précision.

Dans le quatrième chapitre, on propose une amélioration du Map matching particulière par la mise en œuvre d'une architecture multi-modèle dans le pré-traitement de la direction du véhicule. Dans ce contexte, on modélise les différentes manœuvres du véhicule (mouvement en ligne droite, virage à gauche et virage à droite) par des modèles circulaires simples associés à des filtres circulaires. On montre que l'architecture multi-modèle circulaire proposée (notée CIMM) est plus robuste pendant les virages et améliore la précision de l'estimation du cap du véhicule. De plus, les probabilités de chaque modèle sont utilisées pour orienter les particules vers les routes les plus probables.

L'évaluation de ce nouvel algorithme de Map matching est faite dans le cas de la bifurcation en Y et dans le contexte réel avec masquage aléatoire des signaux GPS. Les performances constatées dépassent de loin celles de l'algorithme topologique classique qui repose uniquement sur la position fournie par la navigation par l'estime. On montre dans le cas problématique classique de deux routes parallèles que l'algorithme proposé affecte aux deux routes des probabilités quasi-identiques en plaçant sur chacune un groupe de particules évoluant indépendamment. Ce qui démontre que l'approche proposée permet de prendre en compte les cas ambigus sans mettre en défaut l'algorithme. Ce sont les manœuvres postérieures du véhicule qui lèveront l'ambiguïté et permettront de décider de la viabilité de l'un ou l'autre des deux groupes de particules.

Dans la dernière partie de ce chapitre, les différentes approches de Map matching proposées sont évaluées sur un parcours réel réalisé avec une voiture instrumentée. L'algorithme de Map matching particulière qui utilise un préfiltrage CIMM montre des performances remarquables par rapport aux autres algorithmes en identifiant correctement 70% des segments pendant un masquage des satellites GPS qui dure 14 minutes, soit 99% de la durée du trajet.

On évoque dans l'expérimentation avec les données réelles les limitations de l'approche de Map matching particulière qui se manifeste par la pénurie de particules pour des durées de masquage prolongées pour une trajectoire sans bifurcation. On explique que cette pénurie, qui entraîne le blocage de l'algorithme, provient de plusieurs sources. On cite notamment l'accumulation des erreurs des capteurs proprioceptifs, les erreurs dues à la vectorisation de la carte et à la non prise en considération de la largeur des routes.

Les perspectives des travaux réalisées dans cette thèse portent sur l'amélioration des architectures proposées et leur application dans les systèmes de localisation modernes.

Le préfiltrage multi-modèle IMM circulaire présenté au chapitre quatre peut être amélioré en proposant des modèles plus évolués pour représenter les manœuvres du véhicule. Cela inclut éventuellement l'ajout de la composante vitesse ou l'accélération angulaire et l'emploi de capteurs supplémentaires comme un gyromètre et une centrale inertielle. Il serait ainsi possible de mieux différencier entre un changement de voie, un virage et un rond point. L'architecture IMM peut aussi être étendue à la troisième génération des systèmes multi-modèle à structure variable (VSMM) où l'on peut envisager l'activation et la désactivation des modèles selon l'historique des manœuvres et les informations extraites de la carte. Ceci permettra d'alléger la charge de calcul en préservant un nombre réduit de modèles probables à chaque fois.

La carte numérique constitue également une autre voie de développement qui peut contribuer à l'amélioration du rendement du Map matching particulière. En effet, l'ajout de la largeur des routes et de l'angle d'inclinaison sur l'axe vertical, comme informations supplémentaires pour chaque segment, permettra de propager les particules avec plus de précision et de mitiger les erreurs dues à la carte.

Enfin, la méthode de Map matching particulière proposée peut constituer une bonne alternative pour la localisation des véhicules de tout type dans les zones urbaines ou même dans les zones couvertes, les sous-sols ou les garages à condition qu'une carte des routes soit disponible pour les zones concernées. Une application intéressante de la méthode proposée concerne la localisation dans les zones portuaires ou logistiques, où le véhicule ou l'engin se trouve tantôt à ciel ouvert, tantôt dans un entrepôt ou sous une structure métallique entravant la bonne réception des signaux satellitaires.

Liste des tableaux

2.1	Algorithme du filtre de Kalman	54
3.1	Erreur de positionnement globale en fonction de l'angle entre les routes.	105
3.2	Paramètres des filtres.	106
3.3	Probabilité d'identification en fonction de l'angle entre les routes.	107
3.4	Pourcentage de segments corrects en fonction du pourcentage de points mas- qués (Erreur GPS = 10m)	109
4.1	Erreur de positionnement globale en fonction de l'angle entre les routes.	135
4.2	Probabilité d'identification en fonction de l'angle entre les routes.	139
4.3	Pourcentage de segments corrects en fonction du pourcentage de points mas- qués (Erreur GPS = 10m)	141
4.4	Pourcentage de segments corrects en fonction du pourcentage des points masqués	146
4.5	Distance moyenne entre position GPS et estimée en fonction du pourcentage de points masqués	147
4.6	Pourcentage des réalisations avec pénurie des particules en fonction du pour- centage de points masqués	149

Table des figures

1.1	Problème du Map matching	6
1.2	Matching point-to-point	8
1.3	Calcul de distance entre un point et un segment	9
1.4	Erreur sur le matching point-to-curve	9
1.5	Matching point-to-curve	9
1.6	Problème de la distance entre deux courbes en matching curve-to-curve	11
1.7	Défaut du matching curve-to-curve	11
1.8	Matching curve-to-curve	12
1.9	Distance entre courbes de longueur différente	12
1.10	Surface entourée par deux lignes	13
1.11	Méthode "Area matching"	14
1.12	Algorithme de Greenfeld : Intersections entre ligne GPS et liens	16
1.13	Algorithme de Greenfeld : Sélection d'un lien incorrect sur une jonction	17
1.14	Algorithme de Quddus	18
1.15	Similarité entre le cap du véhicule et la direction du lien	19
1.16	Distance perpendiculaire	20
1.17	Position relative par rapport aux liens reliés à un nœud	21
1.18	Processus de l'algorithme topologique amélioré	23
1.19	Algorithme de Map matching selon l'approche probabiliste	25
1.20	Zone d'erreur autour de la position GPS	26
1.21	Algorithme de Map matching probabiliste	28
1.22	Détermination de la position du véhicule avec la vitesse et le cap	29
1.23	Estimation de la position du véhicule sur le lien	30
1.24	Matching sur une jonction (MPJ)	35
1.25	Système d'inférence flou de Mamdani	37
2.1	Échantillonnage de Monte Carlo selon [75]	55
2.2	Poids des échantillons selon [75]	56

2.3	Structure générale de l'estimation multi-modèles selon [53]	60
2.4	Structure de l'estimateur multi-modèles statique selon [9]	63
2.5	Cycle de l'estimateur multi-modèles autonome	64
2.6	Structure de l'estimateur GPB1 selon [9]	67
2.7	Cycle de l'estimateur GPB1	67
2.8	Structure de l'estimateur GPB2 selon [9]	68
2.9	Cycle de l'estimateur GPB2	69
2.10	Structure de l'estimateur IMM selon [9]	71
2.11	Cycle de l'estimateur IMM	72
3.1	Problème du matching sur une jonction en Y	83
3.2	Direction GPS comparée à la direction réelle	84
3.3	Principe de la localisation par l'estime	85
3.4	Distribution initiale des particules	86
3.5	Particule atteignant la fin d'un segment	87
3.6	Affectation des poids aux particules	88
3.7	Statistiques sur un cercle	91
3.8	Distribution de von Mises pour différents κ et $\mu = \pi$	91
3.9	Comportement du filtre particulaire dans un virage	95
3.10	Architecture du filtre de Map matching	96
3.11	Les trajectoires d'évaluation	102
3.12	Estimation de la direction	103
3.13	Estimation de la direction	104
3.14	Erreur sur la position avec et sans pré-filtrage	105
3.15	Probabilité d'identifier correctement le segment de route	107
3.16	Trajectoire synthétique du véhicule avec masquage GPS	108
3.17	Pourcentage des segments correctement identifiés au sujet de différentes pré- sions GPS pour MAP-TOP, tous les points GPS étant masqués pour MAP-SIR .	109
3.18	Pourcentage des segments correctement identifiés en fonction du pourcentage de points masqués	110
4.1	Distribution de von Mises	122
4.2	Structure du filtre CIMM	125
4.3	Architecture du filtre de Map matching avec pré-filtrage CIMM	130
4.4	Placement des particules en fin de route	131
4.5	Estimation de la direction avec le filtre CIMM	133
4.6	Probabilités des modèles du filtre CIMM	134

4.7	Erreur sur la position : Direction filtrée par le CIMM	135
4.8	Comportement des particules à la bifurcation	136
4.9	Erreur sur la position : expérimentation sans virage	137
4.10	Probabilité d'identifier correctement le segment de route	138
4.11	Probabilité d'identifier correctement le segment de route - Sans virage	139
4.12	Pourcentage des segments correctement identifiés pour différentes précisions GPS	141
4.13	Pourcentage des segments correctement identifiés en fonction du pourcentage de points masqués	142
4.14	Voiture instrumentée utilisée dans l'expérimentation	143
4.15	Trajet réalisé avec la voiture instrumentée	144
4.16	Trajectoire avec données réelles - GPS et estime	144
4.17	Mesures de vitesse, de la direction et probabilités de modèle du CIMM	146
4.18	Pourcentage des segments correctement identifiés en fonction du pourcentage des points masqués - Données réelles	147
4.19	Distance moyenne entre position GPS et estimée en fonction du pourcentage des points masqués - Données réelles	148
4.20	Comportement des algorithmes en cas de routes parallèles	149
4.21	Pourcentage des réalisations avec pénurie de particules en fonction du pourcentage de points masqués - Données réelles	150
4.22	Vectorisation d'un rond-point	151

Bibliographie

- [1] H. Alt, A. Efrat, G. Rote, and C. Wenk. Matching planar maps. *J. of algorithms*. *J. of Algorithms*, 49 :262–283, 2003.
- [2] S.M. Aly, R. El Fouly, and H. Baraka. Extended kalman filtering and interacting multiple model for tracking maneuvering targets in sensor networks. In *Proc. 13th International Conference on Aerospace Sciences and Aviation Technology*, 2009.
- [3] J.M. Anderson and E.M. Mikhail. *Surveying : theory and practice*. WCB/McGraw-Hill, 1998.
- [4] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for on-line nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50 :174–188, 2002.
- [5] A. Averbuch, S. Itzikowitz, and T. Kapon. Radar target tracking-viterbi versus imm. *IEEE Trans. Aerospace and Electronic Systems*, AES-27(3) :550–563, Mai 1991.
- [6] M. Azmani. *Fusion non-linéaire appliquée aux voies pilote et données du signal GPS L2C*. PhD thesis, Dept. Elect. Eng., Université du littoral côte d’Opale, Calais, France, 2010.
- [7] M. Azmani, S. Reboul, J.B. Choquel, and M. Benjelloun. A recursive fusion filter for angular data. In *Proc. IEEE International Conference on Robotics and Biomimetics*, 2009.
- [8] Y. Bar-Shalom and X.R. Li. *Multitarget-Multisensor Tracking : Principles and Techniques*. YBS Publishing : Storrs, 1995.
- [9] Y. Bar-Shalom, X.R. Li, and T. Kibururajan. *Estimation with applications to tracking and navigation : Theory, algorithms and software*. John Wiley and Sons, Inc, 2001.
- [10] Y. Baram and N.R. Sandel. Consistent estimation on finite parameter sets with application to linear systems identification. In *IEEE Transactions Automatic Control*, AC-23(3), pages 451–454, 1978.
- [11] M. Basseville. Detecting changes in signals and systems. *Automatica*, 24(3) :309–326, 1988.

- [12] M. Basseville and I. Nikiforov. *Detection of Abrupt Changes : Theory and Application*. Prentice Hall : Englewood Cliffs, 1993.
- [13] J.L. Bentley and H.A. Mauer. Efficient worst-case data structures for range searching. *Acta Inf.*, 13 :155–168, 1980.
- [14] N. Bergman. *Recursive Bayesian Estimation : Navigation and Tracking Applications*. PhD thesis, Linkoping University, Sweden, 1999.
- [15] D. Bernstein and A. Kornhauser. An introduction to map matching for personal navigation assistants. *New Jersey TIDE Center*, <http://www.njtude.org/reports/mapmatchintro.pdf>, 1996.
- [16] D. Bernstein and A. Kornhauser. Map matching for personal navigation assistants. In *Proceedings of the 77th annual meeting of the Transportation Research Board, Washington D.C.*, pages 11–15, 1998.
- [17] D. Betaille, R. Toledo-Moreo, and J. Laneurit. Making an enhanced map for lane location based services. In *Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on*, pages 711–716, Oct 2008.
- [18] W. D. Blair and B. M. Keel. Radar systems modeling for tracking. *Y. Bar-Shalom, editor, Multitarget-Multisensor Tracking : Applications and Advances, Artech House*, III :321–393, 2000.
- [19] W. D. Blair, G. A. Watson, and S. A. Hoffman. 1994 american control conf. In *Benchmark Problem for Beam Pointing Control of Phased Array Radar Against Maneuvering Target*, pages 2071–2075, 1994.
- [20] H. A. P. Blom. A sophisticated tracking algorithm for ATC surveillance data. In *the International Radar Conference, Paris, France*, 1984.
- [21] H.A.P Blom and Y. Bar-Shalom. Ieee trans. automatic control. In *The interacting multiple model algorithm for systems with markovian switching coefficients*, volume 33 (8), pages 780–783, 1988.
- [22] H.A.P. Blom, R.A. Hogendoorn, and B.A. van Doorn. Design of a multisensor tracking system for advanced air traffic control. *Y. Bar-Shalom, editor, Multitarget-Multisensor Tracking : Applications and Advances, Artech House*, II, 1992.
- [23] S. Brakatsoulas, D. Pfoser, R. Salas, and C. Wenk. On map-matching vehicle tracking data. In *31st VLDB Conference, Trondheim, Norway*, pages 853–864, 2005.
- [24] W. Chen, M. YU, Z.-L. Li, and Y.Q. Chen. Integrated vehicle navigation system for urban applications. *GNSS, Graz*, CD-ROM :15pp, 2003.

-
- [25] A. Doucet, N. De Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer, 2001.
- [26] K. El mokhtari, S. Reboul, M. Azmani, J.B. Choquel, S. Hamdoune, B. Amami, and M. Benjelloun. A circular interacting multi-model filter applied to map matching. In *Proceedings of the 16th International Conference on Information Fusion, Istanbul, Turkey*, pages 1473–1478, 2013.
- [27] N.I. Fisher. *Statistical analysis of circular data*. Cambridge University press, 1993.
- [28] H. Fuchs, Z.M. Kedem, and B.F. Naylor. On visible surface generation by a priori tree structures. *Comput. Graphics*, 14 :124–133, 1980.
- [29] J.S. Greenfeld. Matching GPS observations to locations on a digital map. In *Proceedings of the 81st Annual Meeting of the Transportation Research Board - Washington D.C.*, pages 554–559, 2002.
- [30] A Hassani, A Bertrand, and M. Moonen. Tracking of a rotating object in a Wireless Sensor Network using fuzzy based adaptive IMM filter. In *Communications and Vehicular Technology in the Benelux (SCVT), 2012 IEEE 19th Symposium on*, pages 1–6, 2012.
- [31] J.D. Hol, T.B. Schon, and F. Gustafsson. On resampling algorithms for particle filters. In *Nonlinear Statistical Signal Processing Workshop, 2006 IEEE*, pages 79–82, 2006.
- [32] S.K. Honey, W.B. Zavoli, K.A. Milnes, A.C. Phillips, M.S. White, and G.E. Loughmiller. Vehicle navigational system and method. *United States Patent No. 4796191*, 1989.
- [33] S.R. Jammalamadaka and A. SenGupta. *Topics in Circular Statistics*. World Scientific Publ, New Jersey, 2001.
- [34] J. Joseph and Jr. LaViola. A comparison of unscented and extended kalman filtering for estimating quaternion motion. In *Proc. IEEE American Control Conference*, 2003.
- [35] J. Kacemi. *Fusion d’Informations dans un Système de Positionnement Hybride GPS Multi-Porteuses/Estime*. PhD thesis, Université du Littoral Côte d’Opale, Calais, France, 2006.
- [36] N. Kaempchen, K. Weiss, M. Schaefer, and K.C.J. Dietmayer. Imm object tracking for high dynamic driving maneuvers. In *Proc. IEEE Intell*, pages 825–830, 2004.
- [37] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transaction of the ASME-Journal of Basic Engineering*, pages 35–45, 1960.
- [38] S. Kim., J.-H Kim, and I.-H Hyun. Development of a map matching algorithm for car navigation system using fuzzy q-factor algorithm. In *Proceedings of the World Congress Intelligent Transport System, October, Seoul, Korea*, 1998.
- [39] W. Kim, G Jee, and J. Lee. Efficient use of digital road map in various positioning for ITS. In *IEEE Symposium on Position Location and Navigation, San Deigo, CA*, 2000.

- [40] Y.S. Kim and K.S. Hong. An IMM algorithm for tracking maneuvering vehicles in an adaptive cruise control environment. *International Journal of Control, Automation and Systems*, 2(3) :310–318, 2004.
- [41] D. G. Lainiotis. Optimal adaptive estimation : Structure and parameter adaptation. *IEEE Transactions on Automatic Control*, AC-16 :160–170, 1971.
- [42] J. Li and M. Fu. Research on route planning and map-matching in vehicle GPS/deadreckoning/electronic map integrated navigation system. In *IEEE Proceedings on Intelligent Transportation Systems*, 2, pages 1639–1643, 2003.
- [43] X. R. Li and Y. Bar-Shalom. Mode-set adaptation in multiple-model estimators for hybrid systems. In *1992 American Control Conference, Chicago, IL*, pages 1794–1799, 1992.
- [44] X. R. Li and Y. Bar-Shalom. Multiple-model estimation with variable structure. *IEEE Transactions on Automatic Control*, 41(4) :478–493, 1996.
- [45] X.R. Li. Hybrid estimation techniques. *Control and Dynamic Systems : Advances in Theory and Applications*, 76 :213–287, 1996.
- [46] X.R. Li. Multiple-Model Estimation with Variable Structure - Part II : Model-Set adaptation. *IEEE Transactions on Automatic Control*, 45(11) :2047–2060, 2000.
- [47] X.R. Li. *Multitarget-Multisensor Tracking : Applications and Advances*. Artech House : Boston, 2000.
- [48] X.R. Li. Multiple-Model Estimation with Variable Structure - Part I : Dynamic Models. *IEEE Transactions on Aerospace and Electronics systems*, 39(4) :1333–1364, 2003.
- [49] X.R. Li and Y. Bar-Shalom. A Recursive Multiple Model Approach to Noise Identification. *IEEE Transactions on Aerospace and Electronics systems*, AES-30 :671–684, 1994.
- [50] X.R. Li and V.P. Jilkov. A Survey of Maneuvering Target Tracking - Part I : Dynamic Models. In *2000 SPIE Conf. on Signal and Data Proc. of Small Targets, Orlando, Florida, U.S.A*, volume 4048, pages 212–235, 2000.
- [51] X.R. Li and V.P. Jilkov. A Survey of Maneuvering Target Tracking : Dynamic Models - Part II : Ballistic Target Models. In *2000 SPIE Conf. on Signal and Data Proc. of Small Targets, Sang Diego, California, U.S.A*, volume 4473, pages 559–581, 2001.
- [52] X.R. Li and V.P. Jilkov. A survey of maneuvering target tracking - Part IV : Decision-based methods. In *2002 SPIE Conference on Signal and Data Processing of Small Targets, Orlando, FL*, volume 4728, pages 511–534, 2002.
- [53] X.R. Li and V.P. Jilkov. A Survey of Maneuvering Target Tracking : Dynamic Models - Part V : Multiple-Model Methods. *IEEE Transactions on Aerospace and Electronics systems*, 41(4) :1255–1321, 2005.

-
- [54] X.R. Li and Y. Zhan. Multiple-Model Estimation with Variable Structure - Part V : Likely-Model Set Algorithm. *IEEE Transactions on Aerospace and Electronics systems*, 36(2) :448–466, 2000.
- [55] X.R. Li, Y. M. Zhang, and X. R. Zhi. Multiple-Model Estimation with Variable Structure - Part IV : Design and evaluation of model-group switching algorithm. *IEEE Transactions on Aerospace and Electronics systems*, 35(1) :242–254, 1999.
- [56] X.R. Li, X. R. Zhi, and Y. M. Zhang. Multiple-Model Estimation with Variable Structure - Part III : Model-group switching algorithm. *IEEE Transactions on Aerospace and Electronics systems*, 35(1) :225–241, 1999.
- [57] X. Lianxia, L. Quan, L. Minghua, and L. Zhong. Map matching algorithm and its application. In *Proceedings of the International Conference on Intelligent Systems and Knowledge Engineering (ISKE)*, 2007.
- [58] J.S. Liu and R. Chen. Sequential monte carlo methods for dynamical systems. *Journal of the American Statistical Association*, 93 :1032–1044, 1998.
- [59] Y. Lou, C. Zhang, Y. Zheng, X. Xie, W. Wang, and Y. Huang. Map-matching for low-sampling-rate GPS trajectories. In *ACM GIS '09, Seattle, WA, USA*, 2009.
- [60] D. T. Magill. Optimal adaptive estimation of sampled stochastic. *IEEE Transactions on Automatic Control*, AC-10 :434–439, 1965.
- [61] E.H. Mamdani and S. Assilian. An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies*, 7(1) :1–13, 1975.
- [62] P. S. Maybeck. *Stochastic Models, Estimation and Control, vol. II*. New York : Academic Press, 1982.
- [63] P. S. Maybeck and R. I. Suizu. Adaptive tracker field-of-view variation via multiple model filtering. *IEEE Transactions on Aerospace and Electronic Systems*, AES-21 :529–539, 1985.
- [64] Y. Meng, W. Chen, Y. Chen, and J.C.H. Chao. A simplified map-matching algorithm for in-vehicle navigation unit. *Research Report, Department of Land Surveying and Geoinformatics, Hong Kong Polytechnic University*, 2003.
- [65] G. Nassreddine, F. Abdallah, and T. Dencœux. Map matching algorithm using belief function theory. In *Proceedings of the 11th International Conference on Information Fusion (FUSION '08)*, pages 995–1002, 2008.
- [66] S.H. Noh and T.J. Kim. A comprehensive analysis of map matching algorithms for ITS. *Hongik Journal of Science and Technology*, 9 :303–313, 1998.

- [67] B. Phuyal. Method and use of aggregated dead reckoning sensor and GPS data for map matching. In *Proceedings of the Institute of Navigation (ION) annual conference, 20-27 septembre, 2002*.
- [68] R. R. Pitre, V. P. Jilkov, and X. R. Li. A comparative study of multiple-model algorithms for maneuvering target tracking. In I. Kadar, editor, *Signal Processing, Sensor Fusion, and Target Recognition XIV*, volume 5809 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, pages 549–560, May 2005.
- [69] J. Pyo, D. Shin, and T. Sung. Development of a map matching method using the multiple hypothesis technique. In *IEEE Proceedings on Intelligent Transportation Systems*, pages 23–27, 2001.
- [70] M. A. Quddus. *High Integrity Map Matching Algorithms for Advanced Transport Teleatics Applications*. PhD thesis, Imperial College London UK, 2006.
- [71] M.A. Quddus, R.B. Noland, and W.Y. Ochieng. A high accuracy fuzzy logic based map matching algorithm for road transport. *Journal of Intelligent Transportation Systems*, 10(3) :103–115, 2006.
- [72] C.O. Sayage and B. Moran. Waveform selection for maneuvering targets within an imm framework. *IEEE Trans. Aerosp. Electron. Syst. J.*, 43(3) :1205–1214, 2007.
- [73] J. Simon and J. Jeffrey. A new extension of the kalman uhlmann-filter to nonlinear systems. In *Proc. International Symposium on Aerospace/Defence Sensing, Simulation and Controls*, 1997.
- [74] D. Srinivasan and R.L. Cheu. Development of an improved ERP system using GPS and AI techniques. In *IEEE Proceedings on Intelligent Transportation Systems*, pages 554–559, 2003.
- [75] C. Stachniss. Short introduction to particle filters and monte carlo localization. 2009. <http://ais.informatik.uni-freiburg.de/teaching/ws12/mapping/pdf/slam09-particle-filter-4.pdf>.
- [76] G. Stienne. *Traitements des signaux circulaires appliqués à l’altimétrie par la phase des signaux GNSS*. PhD thesis, Ecole Doctorale S.I., Université du littoral côte d’Opale, Calais, France, 2013.
- [77] G. Stienne, S. Reboul, M. Azmani, J.B. Choquel, and M. Benjelloun. A multi-sensor circular particle filter applied to the fusion of the gps-l2c channels. In *Proc. of the 14th International Conference on Information Fusion*, 2011.
- [78] G. Stienne, S. Reboul, M. Azmani, J.B. Choquel, and M. Benjelloun. A multi-temporal multi-sensor circular fusion filter. In *Information Fusion*, pages 1–15, 2013.

-
- [79] G. Stienne, S. Reboul, M. Azmani, J.B. Choquel, and M. Benjelloun. A multi-temporal multi-sensor circular fusion filter. *Information Fusion*, 18(0) :86–100, 2014.
- [80] G. Stienne, S. Reboul, JB. Choquel, and M. Benjelloun. Circular data processing tools applied to a phase open loop architecture for multi-channels signals tracking. In *Proc. PLANS, Myrtle Beach*, 2012.
- [81] M. Sugeno. *Industrial applications of fuzzy control*. Elsevier Science Pub. Co., 1985.
- [82] S. Syed and M.E. Cannon. Fuzzy logic-based map matching algorithm for vehicle navigation system in urban canyons. In *Proceedings of the Institute of Navigation (ION) national technical meeting, 26-28 January, California, USA*, 2004.
- [83] G. Taylor, G. Blewitt, D. Steup, S. Corbett, and A. Car. Road reduction filtering for GPS-GIS navigation. *Transactions in GIS, ISSN 1361-1682*, 5(3) :193–207, 2001.
- [84] R. Toledo-Moreo, D. Betaille, F. Peyret, and J. Laneurit. Fusing GNSS, dead-reckoning, and enhanced maps for road vehicle lane-level navigation. *Selected Topics in Signal Processing, IEEE Journal of*, 3(5) :798–809, Oct 2009.
- [85] J. K. Tugnait. Adaptive estimation and identification for discrete systems with markov jump parameters. *IEEE Trans. Automatic Control*, AC-27(5) :1054–1065, Oct. 1982.
- [86] K. Watanabe. *Adaptive Estimation and Control : Partitioning Approach*. Prentice Hall : New York, 1992.
- [87] K. Weiss, N. Kaempchen, and A. Kirchner. Multiple-model tracking for the detection of lane change maneuvers. In *Proc. IEEE Intell*, pages 937–942, 2004.
- [88] C.E. White, D. Bernstein, and A.L. Kornhauser. Some map matching algorithms for personal navigation assistants. *Transportation Research Part C*, 8 :91–108, 2000.
- [89] J.S. Yang, K.S. Chon, and S.P. Kang. The map matching algorithm of GPS data with relatively long polling time intervals. *Journal of the Eastern Asia Society for Transportation Studies*, 6 :2561–2573, 2005.
- [90] L.A Zadeh. Fuzzy sets. *Information and Control*, 8 :338–353, 1965.
- [91] L.A. Zadeh. Knowledge representation in fuzzy logic. In *IEEE Transactions on Knowledge and Data Engineering*, 1, pages 89–100, 1989.
- [92] L. Zhao, W.Y. Ochieng, M.A Quddus, and R.B. Noland. An extended kalman filter algorithm for integrating gps and low-cost dead reckoning system data for vehicle performance and emissions monitoring. *The Journal of Navigation*, 56 :257–275, 2003.
- [93] Y. Zhao. *Vehicle Location and Navigation System*. Artech House, Inc., MA., 1997.

Résumé

La navigation dans les environnements contraints tels que les zones portuaires ou les zones urbaines denses est souvent exposée au problème du masquage des satellites GPS. Dans ce cas, le recours aux capteurs proprioceptifs est généralement la solution envisagée pour localiser temporairement le véhicule sur une carte. Cependant, la dérive de ces capteurs met rapidement en défaut le système de navigation. Pour localiser le véhicule, on utilise dans cette thèse, un magnétomètre pour la mesure du cap dans un repère absolu, un capteur de vitesse et une carte numérique du réseau de routes.

Dans ce contexte, le premier apport de ce travail est de proposer la mise en correspondance des mesures de cap avec la carte numérique (map matching) pour localiser le véhicule. La technique proposée fait appel à un filtre particulière défini dans le domaine circulaire et à un préfiltrage circulaire des mesures de cap. On montre que cette technique est plus performante qu'un algorithme de map matching topologique classique et notamment dans le cas problématique d'une jonction de route en Y. Le deuxième apport de ce travail est de proposer un filtre circulaire multi-modèles CIMM défini dans un cadre bayésien à partir de la distribution circulaire de von Mises. On montre que l'intégration de cette nouvelle approche dans le préfiltrage et l'analyse des mesures de cap permet d'améliorer la robustesse de l'estimation de la direction pendant les virages ainsi que d'augmenter la qualité du map matching grâce à une meilleure propagation des particules du filtre sur le réseau de routes. Les performances des méthodes proposées sont évaluées sur des données synthétiques et réelles.

Mots-clés: Map matching, Estimation circulaire, Filtrage particulière, Filtrage multi-modèles.

Abstract

Navigation in constrained areas such as ports or dense urban environments is often exposed to the problem of non-line-of-sight to GPS satellites. In this case, proprioceptive sensors are generally used to temporarily localize the vehicle on a map. However, the drift of these sensors quickly cause the navigation system to fail. To localize the vehicle, a magnetometer is used in this thesis for heading measurement under an absolute reference together with a velocity sensor and a digital map of the road network.

In this context, the first contribution of this work is to provide a matching of the vehicle's heading with the digital map (map matching) to localize the vehicle. The proposed technique uses a particle filter defined in the circular domain and a circular pre-filtering on the heading measurements. It is shown that this technique is more efficient than a conventional topological map matching algorithm, particularly in ambiguous cases like a Y-shape road junction. The second contribution of this work is to propose a circular multiple model filter CIMM defined in a Bayesian framework from the von Mises circular distribution. It is shown that the integration of this new approach in the pre-filtering and analysis of the heading observations improves the robustness of the heading's estimation during cornering and increases the map matching's quality through a better propagation of the particles on the road network. The performances of the proposed methods are evaluated on synthetic and real data.

Keywords: Map matching, Circular estimation, Particle filtering, Multiple Model filtering